

CONSTRUÇÃO DE ALGORITMOS

Bacharelado em Sistemas da Informação

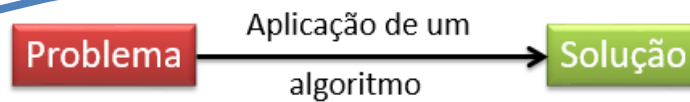
Prof. Marco André Abud Kappel

Aula 2 – Programas de computador

Programas de computador

- Anteriormente:

Hoje:
Programas de
Computador

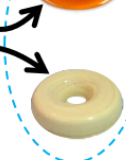
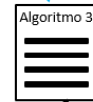
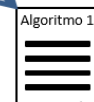
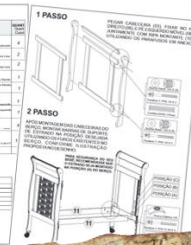


ingredientes

- ✓ 2 xícaras de açúcar
- ✓ 3 xícaras de farinha de trigo
- ✓ 4 colheres de margarina bem cheias
- ✓ 3 ovos
- ✓ 1 1/2 xícara de leite aproximadamente
- ✓ 1 colher (sopa) de fermento em pó bem cheia

modo de preparo

- ✓ Bata as claras em neve
- ✓ Reserve
- ✓ Bata bem as gemas com a margarina e o açúcar
- ✓ Acrescente o leite e farinha aos poucos sem parar de bater
- ✓ Por último agregue as claras em neve e o fermento
- ✓ Coloque em forma grande de furo central untada e enfarinhada
- ✓ Asse em forno médio, pré - aquecido, por aproximadamente 40 minutos
- ✓ Quando espetar um palito e sair limpo estará assado



Programas de computador

- **Introdução**

- Seres **humanos** e **computadores** não falam o mesmo **idioma**.
- Todos os idiomas falados pelas **pessoas** envolvem inúmeros **símbolos** e **combinações** deles.
- O idioma que computadores conseguem entender é a **Linguagem de Máquina**, composta apenas por **dois símbolos**: 0 e 1.



Programas de computador

- Introdução



Programas de computador

- **Introdução**

- De modo simples, podemos entender que os dois símbolos significam que **não há** pulso de energia (0) ou que **há** impulso de energia (1) durante um **intervalo de tempo** (clock) fixado para cada máquina.



- Para o computador, cada **sequencia de zeros e uns** possui um **significado**.
- Para programar, então, precisaríamos **conhecer** estes significados.

Programas de computador

- **Introdução**

- Vamos supor que a seguinte tabela apresenta os **comandos** que algumas sequencias binárias representam:

Comando / símbolo	Código binário
Escrever na tela	00001111
Letra “o”	10010110
Letra “i”	01101001

- Como poderíamos, então, **pedir** para a máquina **escrever** na tela a palavra “oi”?

Programas de computador

- **Introdução**

- Vamos supor que a seguinte tabela apresenta os **comandos** que algumas sequencias binárias representam:

Comando / símbolo	Código binário
Escrever na tela	00001111
Letra “o”	10010110
Letra “i”	01101001

- Como poderíamos, então, **pedir** para a máquina **escrever** na tela a palavra “oi”?

000011111001011001101001

Programas de computador

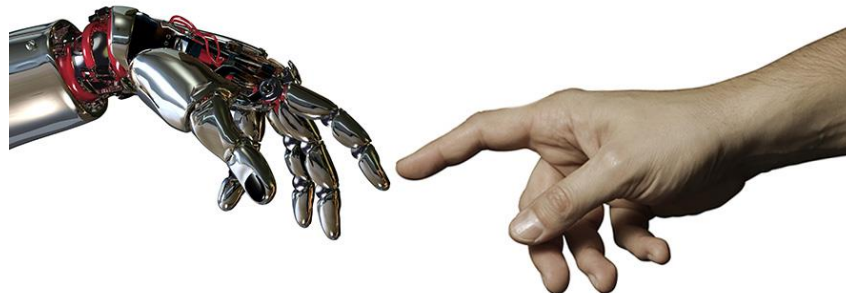
- **Introdução**

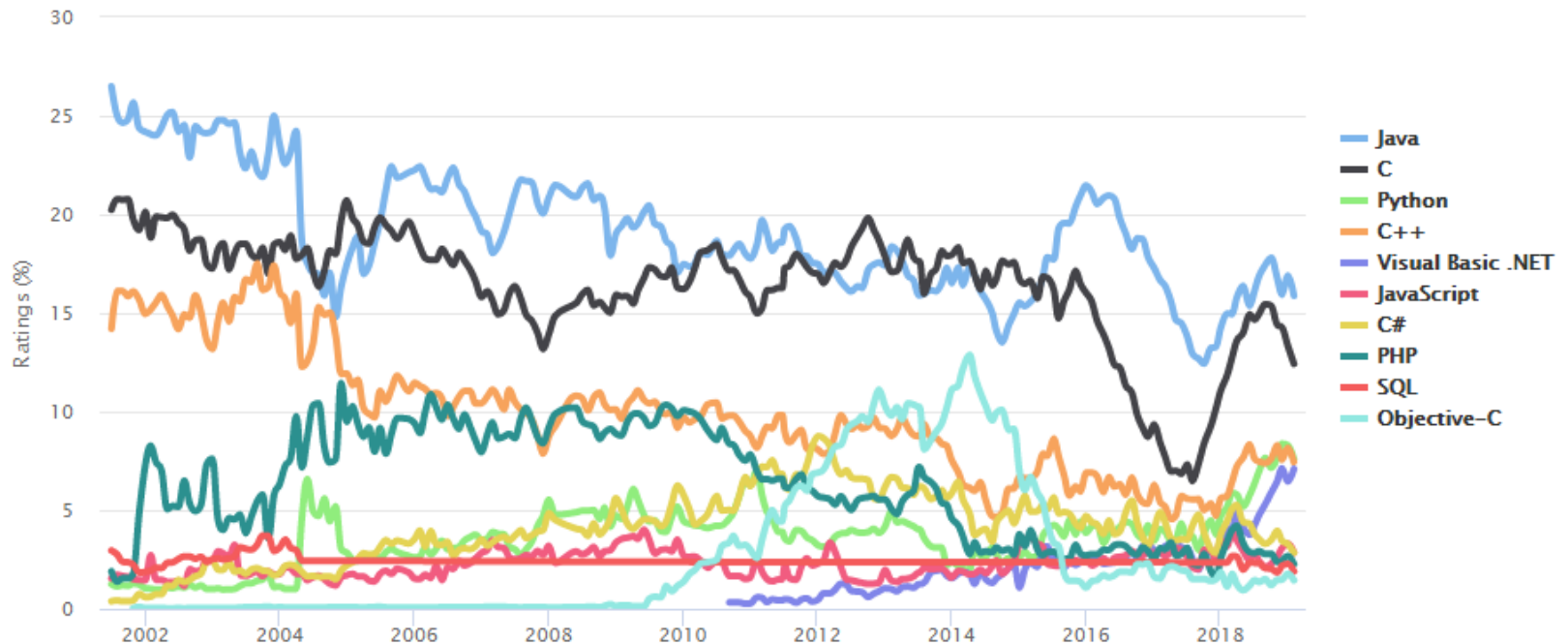
- Com isso, é possível notar algumas **dificuldades** para programar:
 - Seria necessário saber qual é o **código binário** referente a cada **comando** e **símbolo**.
 - Cada **hardware diferente** possui um **conjunto próprio** de significados para diferentes sequencias binárias.
 - Todos os programas seriam **enormes** e **nada intuitivos** para a compreensão dos **seres humanos**.
- Felizmente, existe uma alternativa...

Programas de computador

- **Linguagens de programação**

- As linguagens de programação surgiram com o objetivo de **facilitar a vida** dos programadores.
- Cada linguagem possui o seu próprio **conjunto de palavras** e uma **sintaxe** própria.
- Porém, todas possuem o **mesmo objetivo**: possibilitar que o **programador** seja capaz de **transmitir algoritmos** para o **computador**.





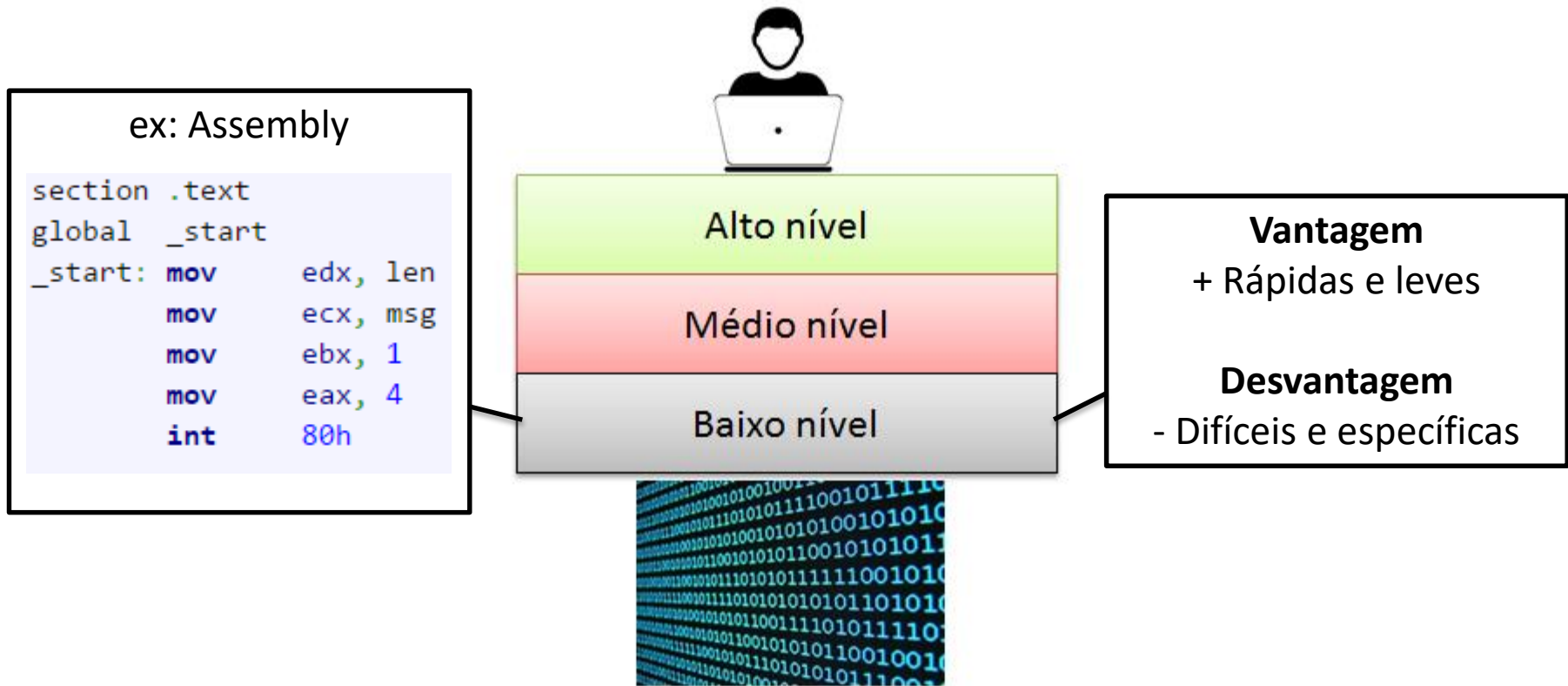
Programas de computador

- **Linguagens de programação**
 - Classificação de linguagens quanto ao nível:



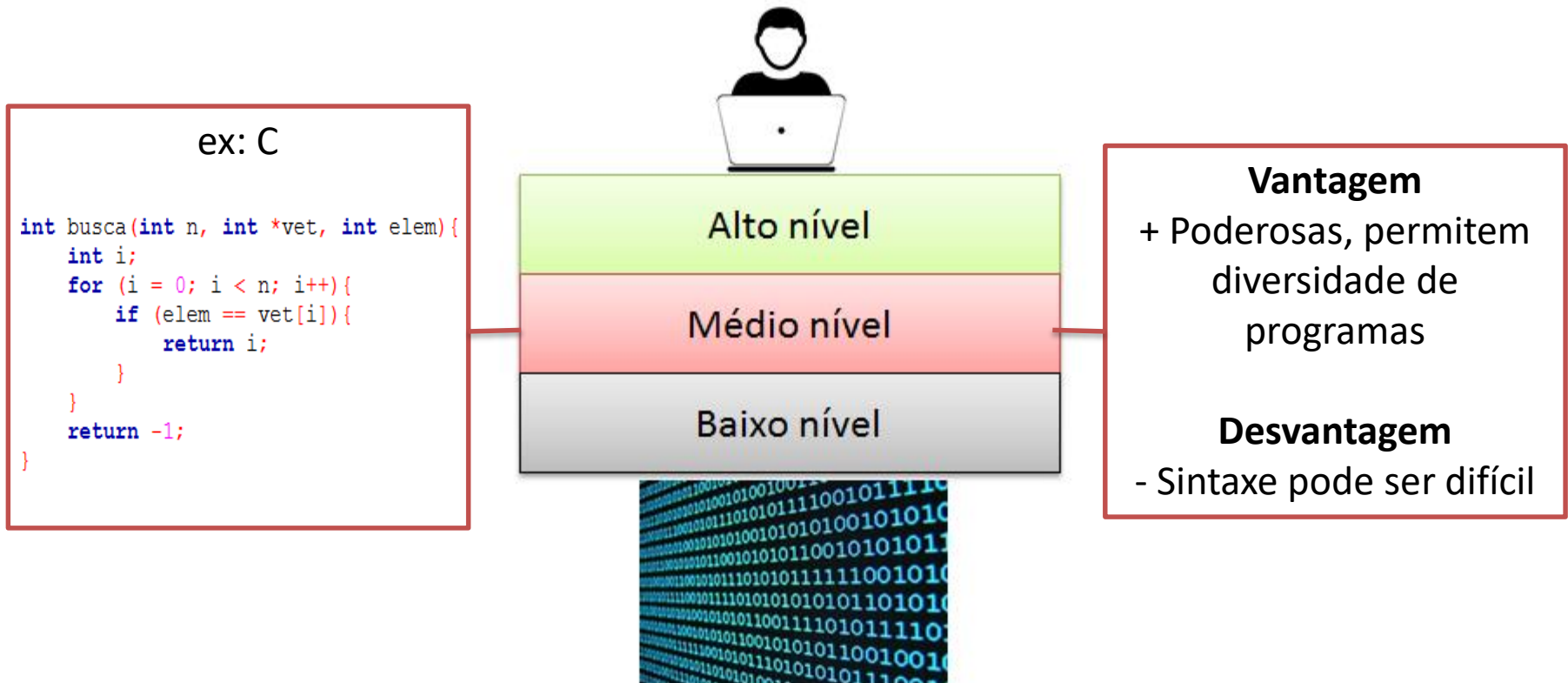
Programas de computador

- **Linguagens de programação**
 - Classificação de linguagens quanto ao nível:



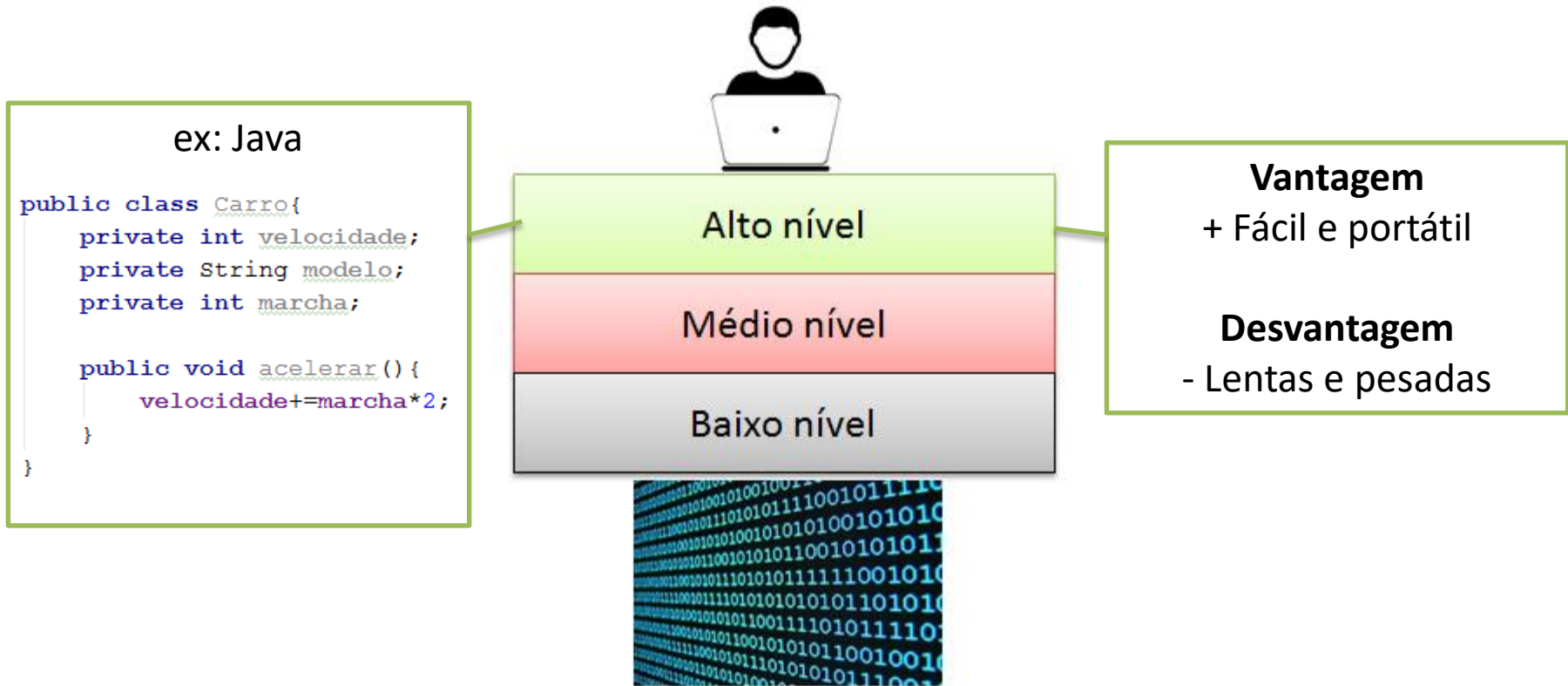
Programas de computador

- **Linguagens de programação**
 - Classificação de linguagens quanto ao nível:



Programas de computador

- Linguagens de programação
 - Classificação de linguagens quanto ao nível:



Programas de computador

- **Linguagens de programação**
 - Classificação de linguagens quanto ao nível:



Mas, se o computador só entende a linguagem de máquina, pra que serve uma linguagem de programação?

ex:

```
public class C  
{  
    private int i;  
    private String s;  
    private int j;  
  
    public void m()  
    {  
        velocidade;  
    }  
}
```

gem
portátil

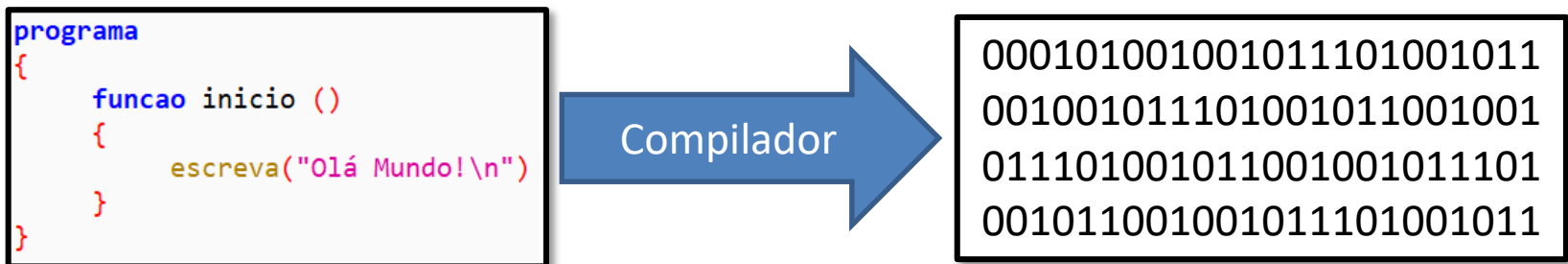
tagem
pesadas



Programas de computador

- **Compilação**

- Cada linguagem de programação oferece um módulo chamado **compilador** (interpretador em alguns casos, mas não se preocupe com isso agora).
- O **objetivo** do compilador é transformar o seu **texto** (escrito na linguagem de programação) em um **arquivo** na linguagem que a máquina **compreenda**.



Programas de computador

- **Compilação**

- O compilador verifica se o programa escrito pelo programador está **corretamente escrito**, considerando as palavras que a linguagem de programação oferece.

```
programa
{
    funsao inicio ()
    {
        iscreva(olá)
    }
}
```

- Se algum **erro** for detectado na verificação, o compilador **reporta** este erro ao programador.

Programas de computador

- **Compilação**

- Se o programa estiver correto, ele traduz a linguagem usada para a **linguagem de máquina**, gerando um arquivo **executável**.

```
programa
{
    funcao inicio()
    {
        escreva("Olá Mundo!\n")
    }
}
```



- A etapa de **compilação** está terminada, agora falta apenas a **execução**.

Programas de computador

- **Execução**

- Para executar o programa, basta clicar duas vezes no **arquivo executável**, ou usar uma linha de comando no prompt de comando do sistema operacional.
- Para aplicações simples, um editor como o **Notepad** é o suficiente para **implementar** um programa.
- Na prática, quando um programador estiver desenvolvendo um programa, ele estará usando um **ambiente de desenvolvimento integrado** (IDE - *Integrated Development Environment*)
- Normalmente, nas IDEs, a execução é feita por um simples apertar de um botão “play”.



Programas de computador

- **Execução**

- **IDEs** fornecem ferramentas que dão **suporte** ao processo de desenvolvimento de software.
 - **Atalhos e ferramentas** para criação/edição de código
 - Indicam **erros de sintaxe**
 - Possuem **depuradores** para investigação de **erros de lógica**
 - Dentre inúmeras outras **utilidades**



Programas de computador

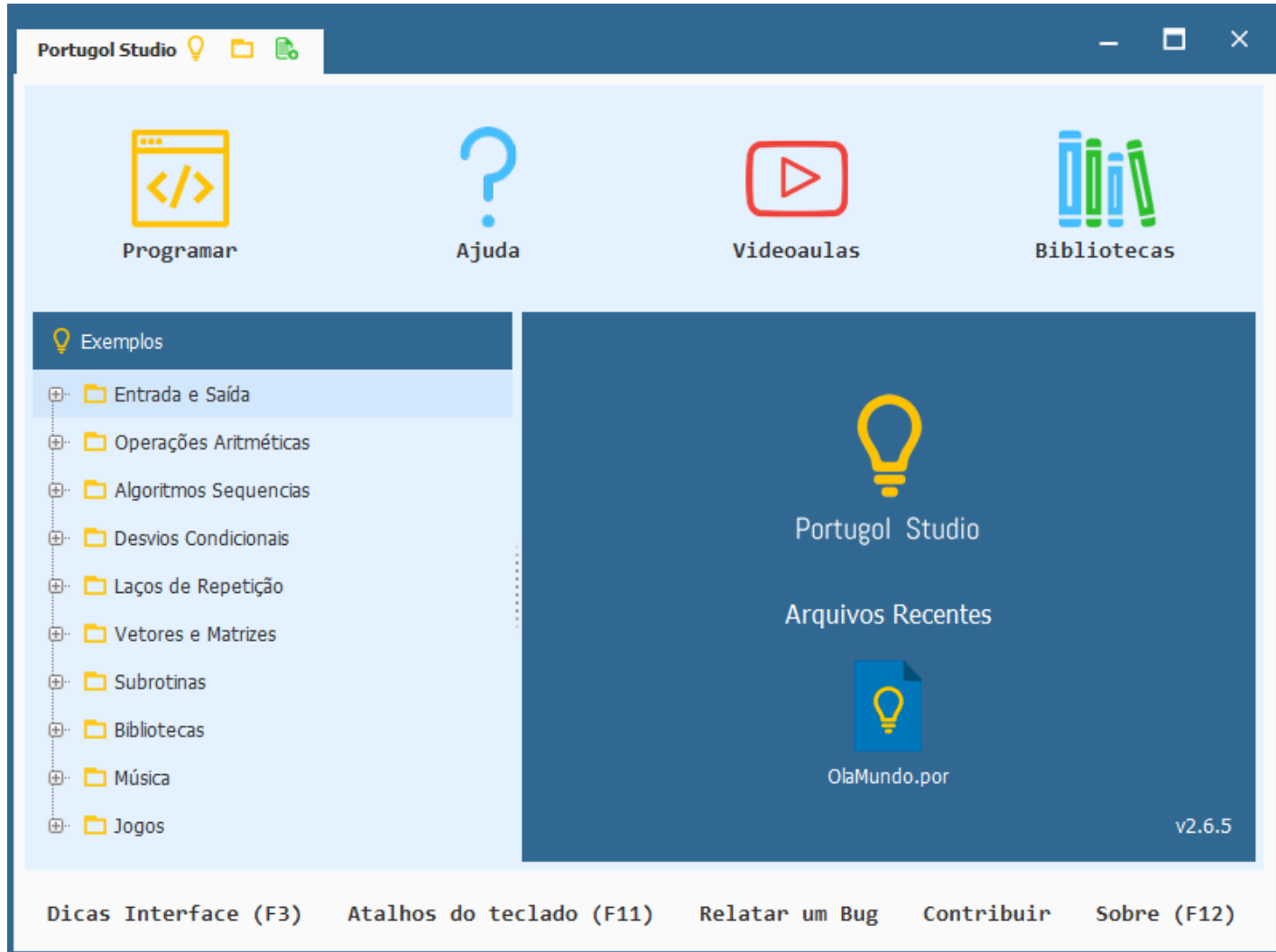
- **Português estruturado**

- A linguagem que vamos abordar no curso é o **Português Estruturado** (Portugol).
- Na verdade, Portugol é uma **pseudolinguagem** que permite que o programador desenvolva algoritmos estruturados em português de forma **simples e intuitiva**.
- É bastante utilizado para **ensinar** lógica de programação e **algoritmos** de maneira fácil e didática para nativos da língua portuguesa.



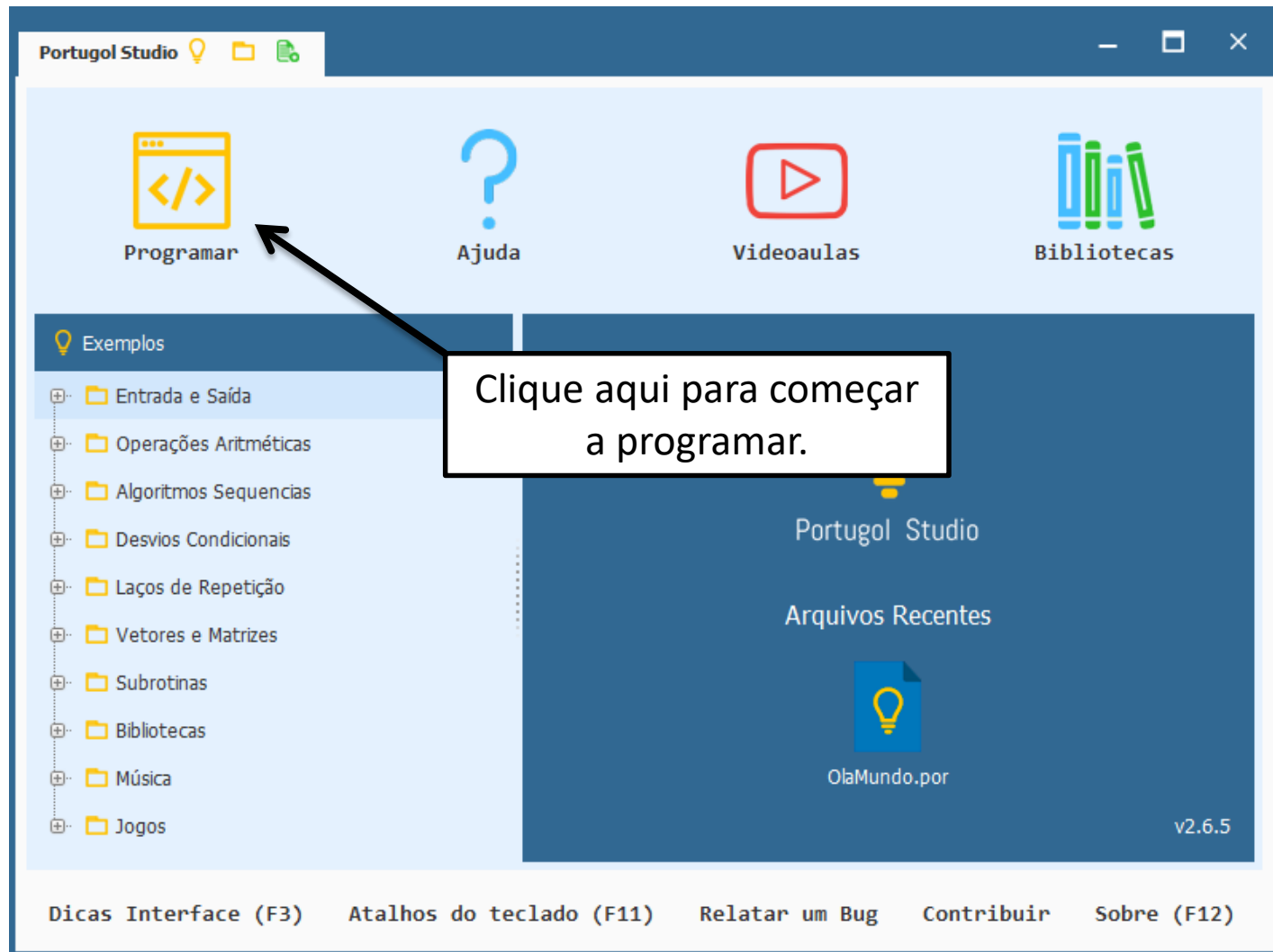
- **Português estruturado**

- O Portugol Studio é o ambiente que vamos utilizar para criar nossos programas.



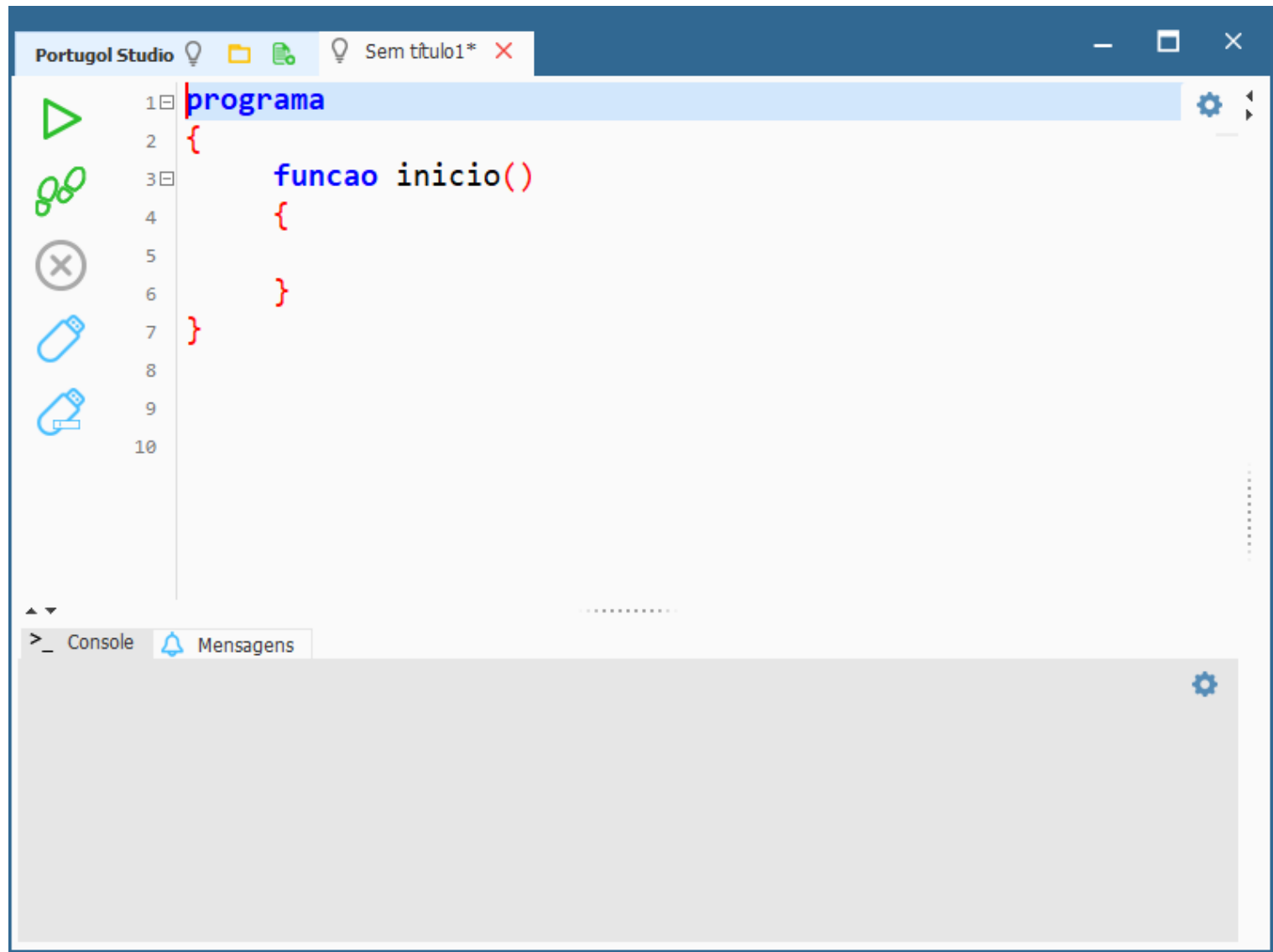
- **Português estruturado**

- O Portugol Studio é o ambiente que vamos utilizar para criar nossos programas.



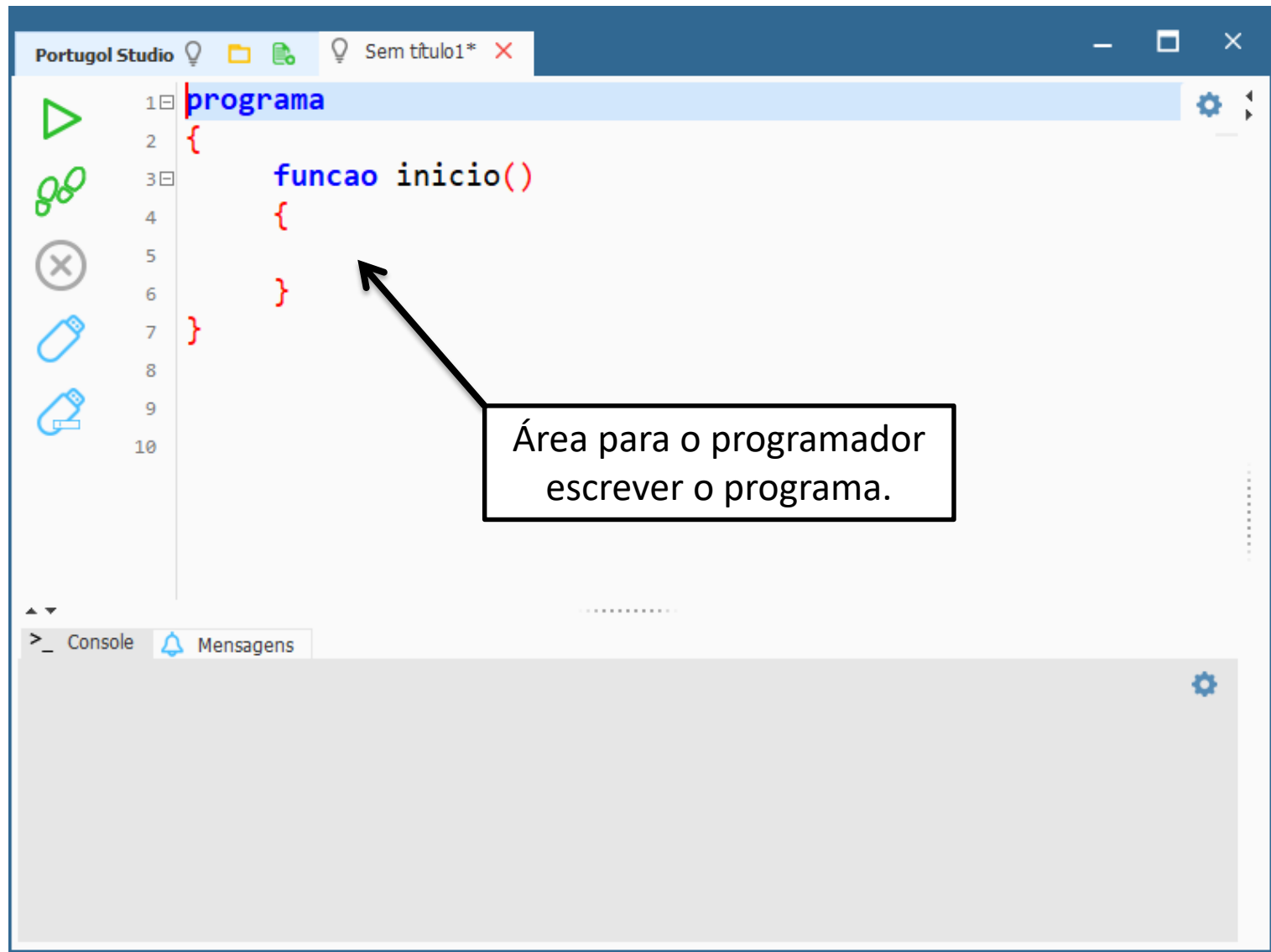
- **Português estruturado**

- Esta é a tela básica do Portugol Studio.



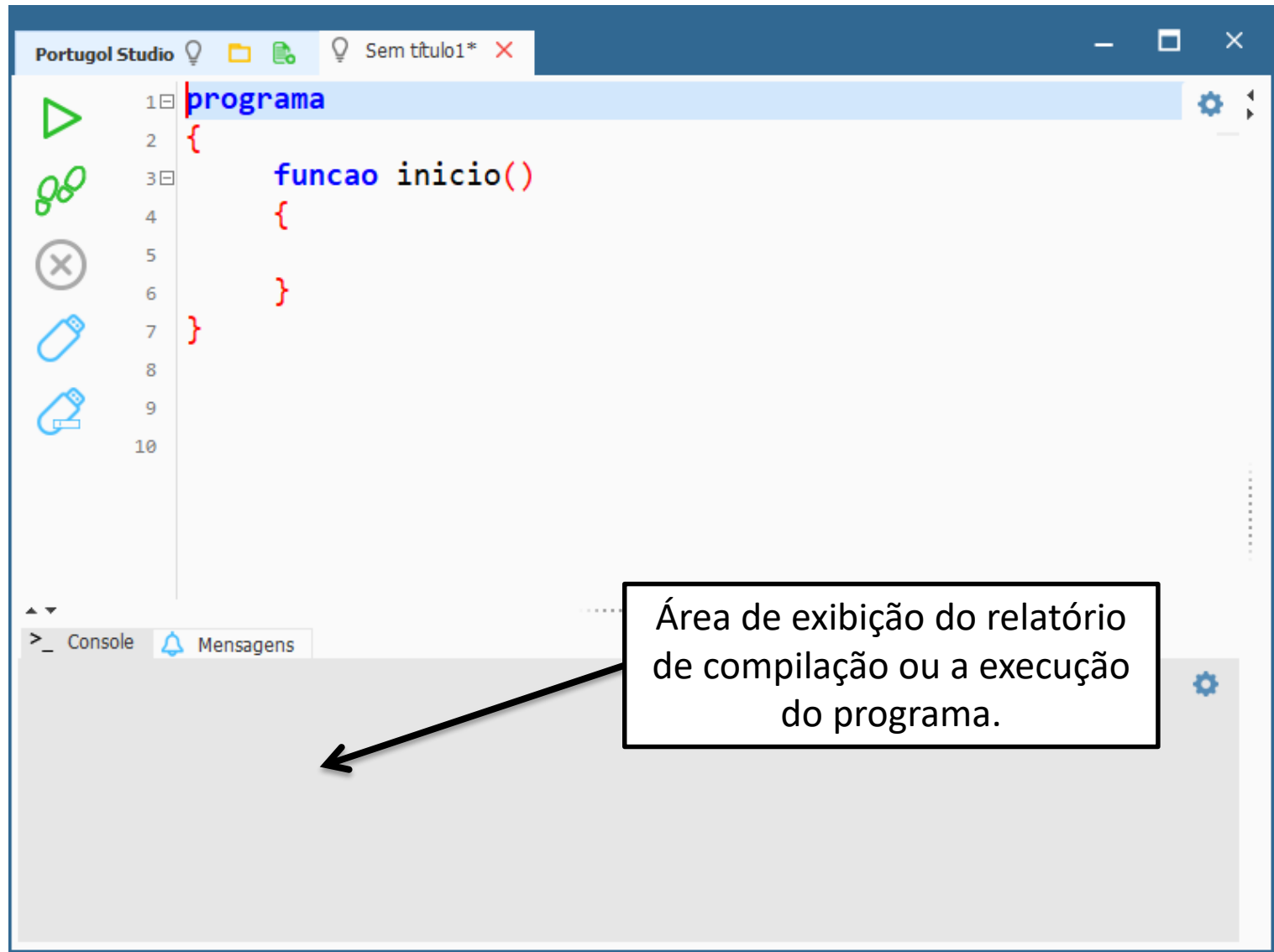
- **Português estruturado**

- Esta é a tela básica do Portugol Studio.



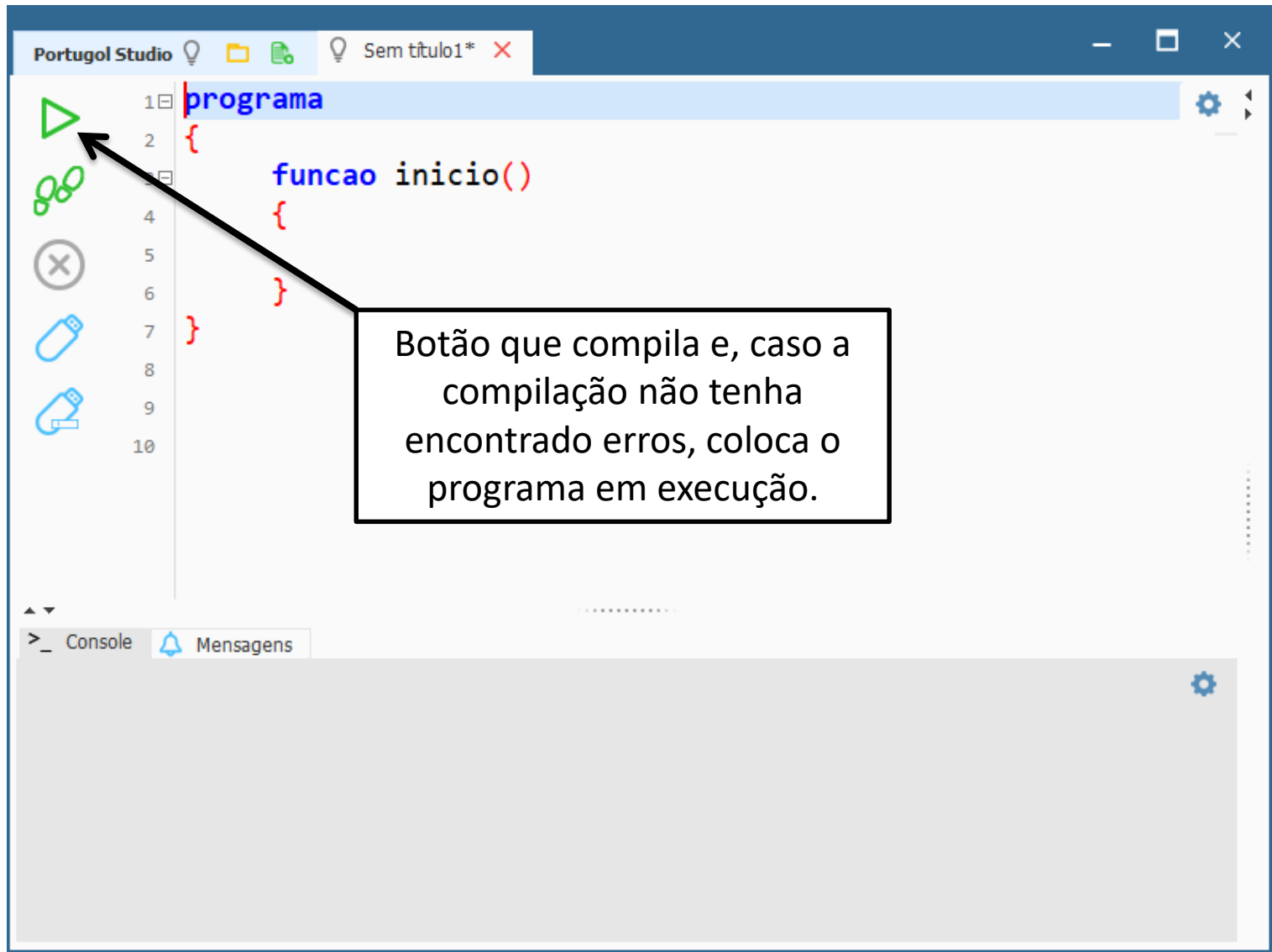
- **Português estruturado**

- Esta é a tela básica do Portugol Studio.



- **Português estruturado**

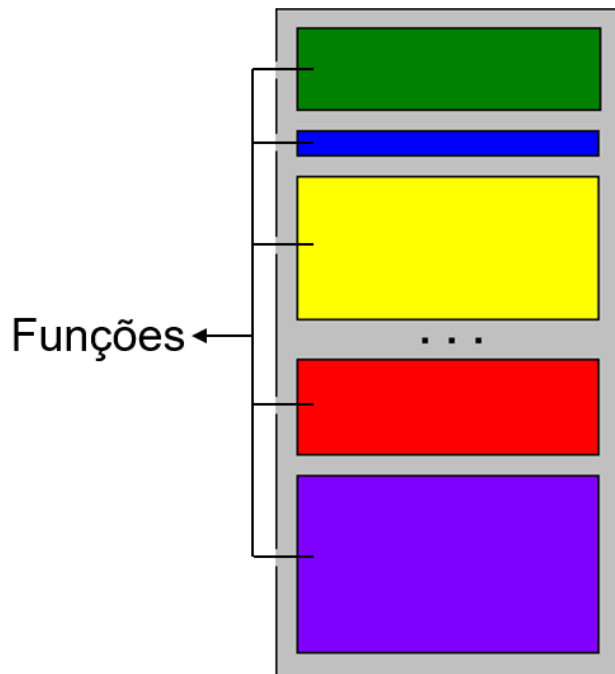
- Esta é a tela básica do Portugol Studio.



Programas de computador

- **Primeiro programa**

- Um programa é um **conjunto de códigos** divididos em várias **funções**.



- Cada uma dessas funções terá um escopo próprio, delimitado pelos símbolos { e }.
- O programa utilizará estas funções para executar um algoritmo.

Programas de computador

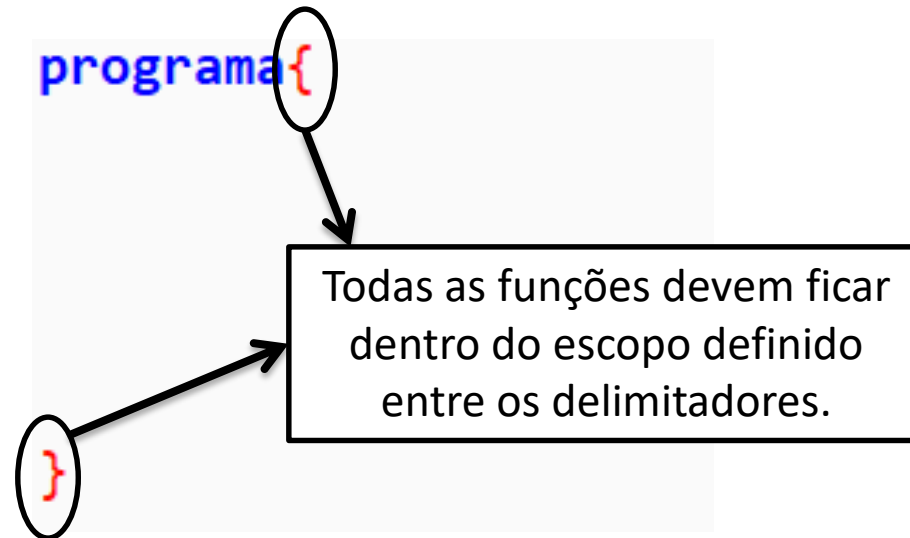
- **Primeiro programa**
 - O **escopo global**, que envolve todo o programa, deve ser indicado com a palavra “programa” e os delimitadores:

```
programa{  
  
  
  
  
  
  
  
  
}
```

Programas de computador

- Primeiro programa

- O **escopo global**, que envolve todo o programa, deve ser indicado com a palavra “programa” e os delimitadores:



Programas de computador

- **Primeiro programa**

- Existe uma **função especial** que precisa estar presente em todos os programas: a função **inicio**.

```
programa{  
  
    funcao inicio(){  
  
    }  
  
}
```

- Esta será a **primeira função** a ser executada quando o programa entrar em execução.

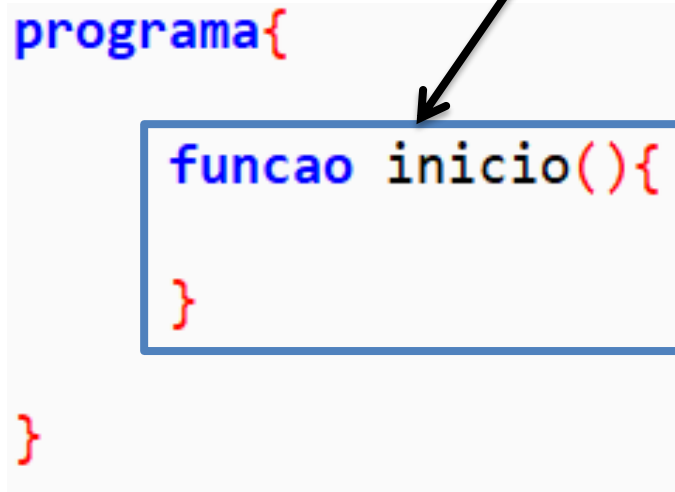
Programas de computador

- **Primeiro programa**

- Existe uma **função especial** que precisa estar presente em todos os programas: a função **inicio**.

Todos os programas
começarão a ser
executados por aqui.

```
programa{  
    funcao inicio(){  
    }  
}
```



- Esta será a **primeira função** a ser executada quando o programa entrar em execução.

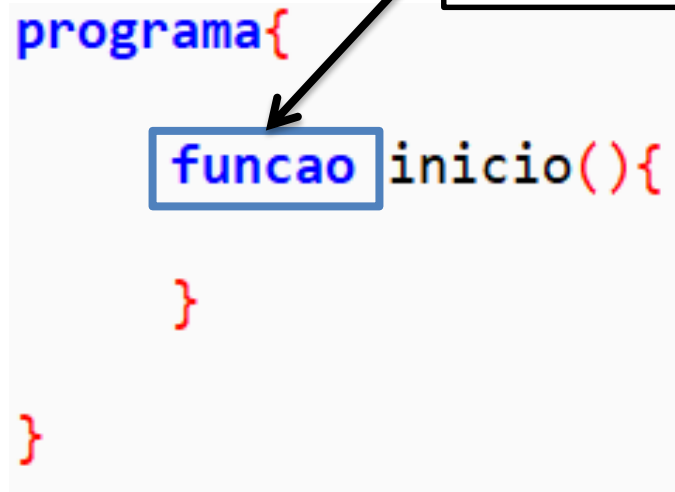
Programas de computador

- Primeiro programa

- Existe uma **função especial** que precisa estar presente em todos os programas: a função **inicio**.

Palavra que indica o início de uma função.

```
programa{  
    funcao inicio(){  
    }  
}
```



- Esta será a **primeira função** a ser executada quando o programa entrar em execução.

Programas de computador

- **Primeiro programa**

- Existe uma **função especial** em todos os programas: a função **inicio**.

Palavra que indica o nome da função. No caso, é a função especial de nome "inicio".

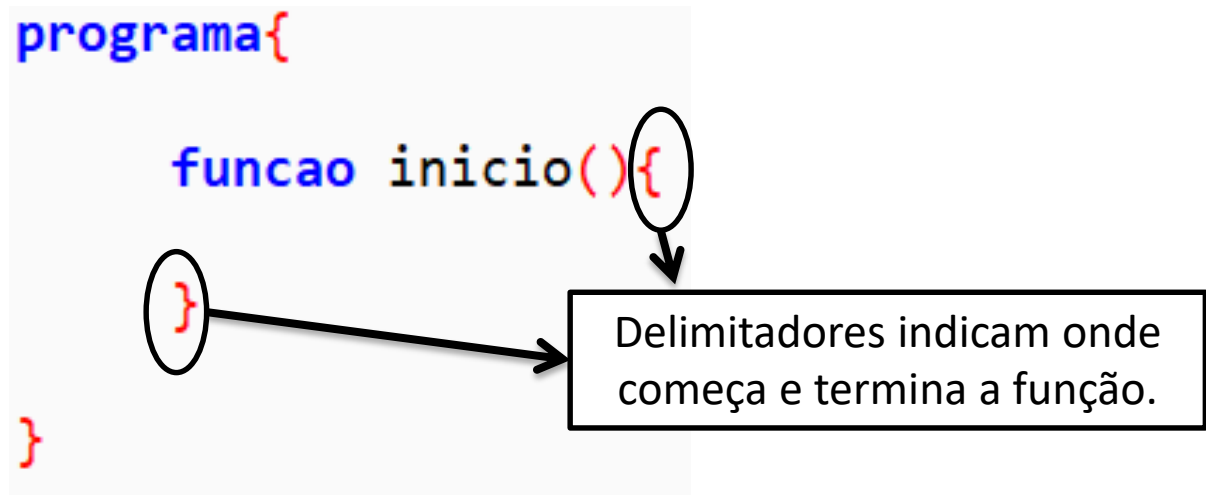
```
programa{  
    funcao inicio(){  
    }  
}
```

- Esta será a **primeira função** a ser executada quando o programa entrar em execução.

Programas de computador

- **Primeiro programa**

- Existe uma **função especial** que precisa estar presente em todos os programas: a função **inicio**.



- Esta será a **primeira função** a ser executada quando o programa entrar em execução.

Programas de computador

- **Primeiro programa**

- Por enquanto, nossos programas serão feitos **apenas** com a função início, mas tenha em mente que várias **outras funções** podem ser criadas no programa.

```
programa {  
  
    funcao inicio(){  
  
    }  
  
    funcao outraFuncao(){  
  
    }  
  
    funcao maisUmaFuncao(){  
  
    }  
  
}
```

Programas de computador

- Primeiro programa

- O programa abaixo, da forma que está, já é um programa **válido** e **executável**.
- Porém, ele **não faz nada**, pois **não há código** inserido dentro do **escopo da função** do programa.

```
programa{  
  
    funcao inicio(){  
  
    }  
  
}
```

Programas de computador

- Primeiro programa

- O programa abaixo, da forma que está, já é um programa **válido** e **executável**.
- Porém, ele **não faz nada**, pois **não há código** inserido dentro do **escopo da função** do programa.

```
programa{  
  
    funcao inicio(){  
  
    }  
  
}
```

– Execute!



Programas de computador

- Primeiro programa

- O programa abaixo, da forma que está, já é um programa **válido** e **executável**.
- Porém, ele **não faz nada**, pois **não há código** inserido dentro do **escopo da função** do programa.

```
programa{  
  
    funcao inicio(){  
  
    }  
  
}
```

– Execute!



> _ Console  Mensagens

Programa finalizado. Tempo de execução: 0 milissegundos

Programas de computador

- A função “escreva”

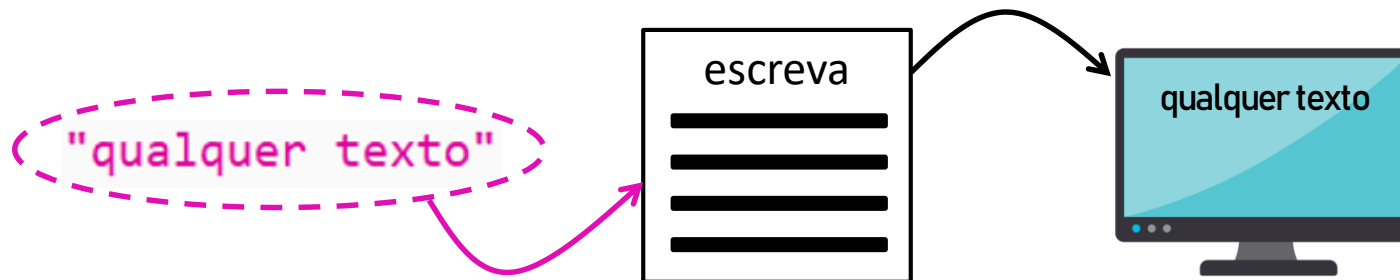
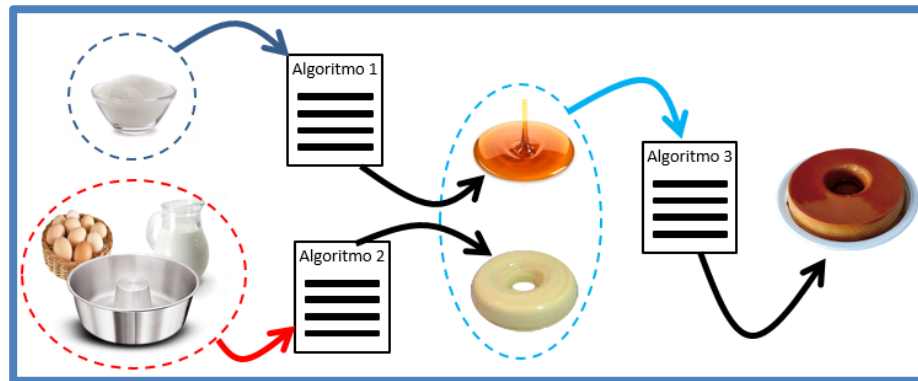
- Podemos, agora, começar a dar **ordens** ao computador.
- Vamos começar mandando-o **escrever um texto** em sua tela.
- O **comando** que faz isso é a função “escreva”.
- A forma de chamar a função “escreva” é justamente escrever o seu nome e estabelecer um **argumento**, o texto que desejamos ser escrito.

```
escreva("qualquer texto")
```

➤ **Obs:** um texto precisa sempre ser escrito entre **aspas**.

Programas de computador

- A função “escreva”



```
escreva("qualquer texto")
```

Programas de computador

- A função “escreva”
 - Implemente, salve e execute o seguinte programa que imprime o texto “Olá mundo!” na tela:

```
programa {  
  
    funcao inicio(){  
        escreva("Olá mundo!")  
    }  
  
}
```

Programas de computador

- A função “escreva”
 - Implemente, salve e execute o seguinte programa que imprime o texto “Olá mundo!” na tela:

```
programa {  
  
    funcao inicio(){  
        escreva("Olá mundo!")  
    }  
  
}
```

> _ Console  Mensagens

Olá mundo!
Programa finalizado. Tempo de execução: 13 milissegundos

Programas de computador

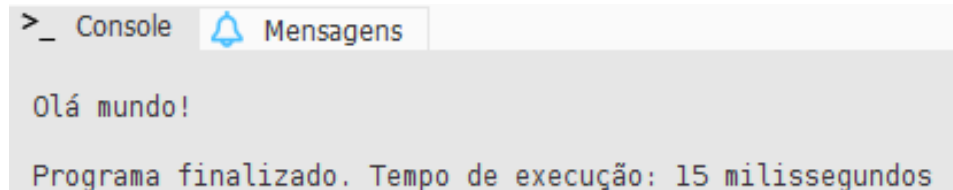
- A função “escreva”

- No exemplo, estamos pedindo para a função “escreva” escrever na tela um **texto constante**, passado como argumento.


```
escreva("Olá mundo!")
```

- Se quisermos que o texto tenha uma **quebra de linha**, basta inserir os caracteres `\n`.

```
programa {  
  
    funcao inicio(){  
        escreva("Olá mundo!\n")  
    }  
  
}
```



The screenshot shows a console window with two tabs: "Console" and "Mensagens". The "Console" tab is active and displays the output "Olá mundo!". Below the output, a status message reads "Programa finalizado. Tempo de execução: 15 milissegundos".

```
>_ Console  Mensagens  
  
Olá mundo!  
  
Programa finalizado. Tempo de execução: 15 milissegundos
```

Programas de computador

- **Exercício**

1. Escreva um programa para escrever no monitor do computador o seguinte texto, preenchido com suas informações:

----- Informações do aluno -----

Nome:

Data de nascimento:

E-mail:

Curso:

Matricula:

Disciplina:

Programas de computador

- **Continuando**

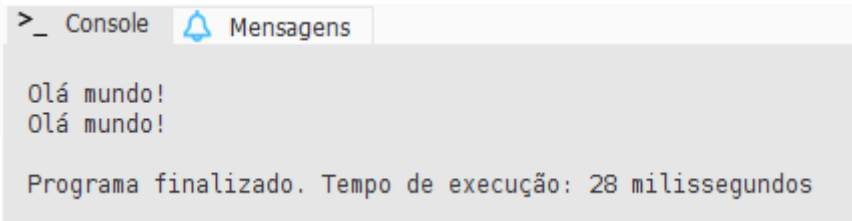
- Até aqui, estamos apenas escrevendo na tela um **texto constante**.
- Se quisermos escrever **duas vezes** o mesmo texto na tela, como faríamos isso?

Programas de computador

- **Continuando**

- Até aqui, estamos apenas escrevendo na tela um **texto constante**.
- Se quisermos escrever **duas vezes** o mesmo texto na tela, como faríamos isso?
- A solução mais simples poderia ser:

```
programa {  
  
    funcao inicio(){  
        escreva("Olá mundo!\n")  
        escreva("Olá mundo!\n")  
    }  
  
}
```



The screenshot shows a console window with two tabs: 'Console' and 'Mensagens'. The 'Console' tab is active and displays the output of the program: 'Olá mundo!' on the first line and 'Olá mundo!' on the second line. Below the output, a status message reads: 'Programa finalizado. Tempo de execução: 28 milissegundos'.

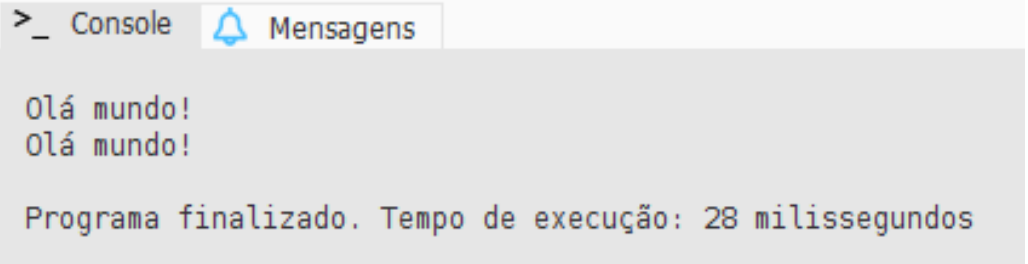
```
> _ Console Mensagens  
  
Olá mundo!  
Olá mundo!  
  
Programa finalizado. Tempo de execução: 28 milissegundos
```

Programas de computador

- **Continuando**

- A função “escreva” ainda possibilita que você escreva na tela vários textos de uma vez só, passando **vários textos como argumento**:

```
programa {  
  
    funcao inicio(){  
        escreva("Olá mundo!\n","Olá mundo!\n")  
    }  
}
```



The screenshot shows a console window with two tabs: "Console" and "Mensagens". The "Console" tab is active and displays the output of the program. The output consists of two lines, "Olá mundo!", each on a new line, followed by a message indicating the program has finished execution and the time taken.

```
>_ Console Mensagens  
  
Olá mundo!  
Olá mundo!  
  
Programa finalizado. Tempo de execução: 28 milissegundos
```


Programas de computador

- **Continuando**

- Essa ainda não parece ser uma boa solução, pois envolve CTRL-C/CTRL-V.
- Muitas vezes, durante o desenvolvimento de programas, precisaremos **guardar valores** para serem usados em outra ocasião.
- Para isso, precisamos utilizar a **memória do computador** para **armazenar dados**.



Programas de computador

- **Variáveis**

- Uma **variável** é uma **posição de memória** reservada para armazenar um valor.
- O **valor** da variável pode **mudar** ao longo da execução de um programa.

“algum texto”



Programas de computador

- **Variáveis**
 - Podemos imaginar que a memória é uma imensa rua e que cada **endereço de memória** é uma casa onde podemos guardar uma informação.
 - Nesse sentido, o programa pode “alugar uma casa” criando uma variável.
 - Cada variável é referente a um **único endereço de memória**.



Programas de computador

- **Variáveis**

- Cada variável precisa ter um **nome**.
- O nome precisa ser um **identificador válido**.
- Este nome irá possibilitar que o **valor** guardado nela seja **modificado** ou **utilizado**.
- O **valor pode mudar**, mas o **tipo de valor** deve sempre ser o **mesmo**.
- **Por exemplo**: uma variável que armazena **números inteiros** não poderá armazenar **textos** ou números com **casas decimais**.



Programas de computador

- **Tipos de dados**

- A tabela a seguir estabelece os **tipos de dados** com os quais podemos trabalhar:

Tipo	Alguns valores possíveis
caracter	'a' 'Z' ' ' '+'
cadeia	"carro" "zero grau" "Rio de Janeiro"
inteiro	-32767 0 26 32767
real	-1.45 0.05 278.90
logico	verdadeiro falso

- Uma vez **definido** o tipo de uma variável, ele **não irá mais mudar**.

Programas de computador

- **Identificadores**

- Um **identificador** de uma variável é um **nome** que damos à posição de memória que será alocada para guardarmos um dado.
- Um **identificador** precisa seguir algumas regras:
 1. O **primeiro caractere** de um identificador só pode ser uma **letra** (sem acento e sem cedilha) ou o **caractere sublinha**(_).
 2. Os **demais caracteres** de um identificador podem ser uma **letra** (sem acento e sem cedilha) ou o **caractere sublinha** ou um **algarismo**.
 3. O identificador **não** pode ser uma **palavra reservada**.

Programas de computador

- **Id** Uma **palavra reservada** é uma palavra que **faz parte** da própria linguagem de programação.

➤ Algumas palavras reservadas do Portugol:

programa	caracter
funcao	cadeia
escreva	logico
inteiro	verdadeiro
real	falso

Essas palavras não podem ser usadas em identificadores.

(sem acento e sem cedilha) ou o **caractere sublinha** ou um **algarismo**.

3. O identificador **não** pode ser uma **palavra reservada**.

Programas de computador

- Declaração de variáveis

- Para **criar** uma variável, é necessário fazer uma **declaração**.
- A **sintaxe** para a declaração é:

tipo **identificador**

Um dos tipos possíveis:

caracter **cadeia** **inteiro**
real **logico**

Qualquer identificador
seguindo as **3 restrições**.

Programas de computador

- **Declaração de variáveis**

- Para **criar** uma variável, é necessário fazer uma **declaração**.
- A **sintaxe** para a declaração é:

tipo **identificador**

- Também é possível declarar na **mesma linha** várias variáveis com o mesmo tipo:

tipo **identificador1, identificador2**

Programas de computador

- **Exemplo:**

- Considere que queremos fazer um programa que vai utilizar variáveis para guardar as seguintes informações:

Tipos possíveis:

caracter cadeia inteiro
real logico

Variável	Valores possíveis	Melhor tipo
dia	1; 3; 31	
mês	3; 7; 12	
ano	1992; 2008	
salário	1320,00	
sexo	'M' ou 'F'	
nome	"Maria"; "João"	
É maior de idade	Verdadeiro ou Falso	

- Quais são os melhores tipos para estas variáveis?

Programas de computador

- **Exemplo:**

- Considere que queremos fazer um programa que vai utilizar variáveis para guardar as seguintes informações:

Tipos possíveis:

caracter cadeia inteiro
real logico

Variável	Valores possíveis	Melhor tipo
dia	1; 3; 31	inteiro
mês	3; 7; 12	
ano	1992; 2008	
salário	1320,00	
sexo	'M' ou 'F'	
nome	"Maria"; "João"	
É maior de idade	Verdadeiro ou Falso	

- Quais são os melhores tipos para estas variáveis?

Programas de computador

- **Exemplo:**

- Considere que queremos fazer um programa que vai utilizar variáveis para guardar as seguintes informações:

Tipos possíveis:

caracter cadeia inteiro
real logico

Variável	Valores possíveis	Melhor tipo
dia	1; 3; 31	inteiro
mês	3; 7; 12	inteiro
ano	1992; 2008	
salário	1320,00	
sexo	'M' ou 'F'	
nome	"Maria"; "João"	
É maior de idade	Verdadeiro ou Falso	

- Quais são os melhores tipos para estas variáveis?

Programas de computador

- **Exemplo:**

- Considere que queremos fazer um programa que vai utilizar variáveis para guardar as seguintes informações:

Tipos possíveis:

caracter cadeia inteiro
real logico

Variável	Valores possíveis	Melhor tipo
dia	1; 3; 31	inteiro
mês	3; 7; 12	inteiro
ano	1992; 2008	inteiro
salário	1320,00	
sexo	'M' ou 'F'	
nome	"Maria"; "João"	
É maior de idade	Verdadeiro ou Falso	

- Quais são os melhores tipos para estas variáveis?

Programas de computador

- **Exemplo:**

- Considere que queremos fazer um programa que vai utilizar variáveis para guardar as seguintes informações:

Tipos possíveis:

caracter cadeia inteiro
real logico

Variável	Valores possíveis	Melhor tipo
dia	1; 3; 31	inteiro
mês	3; 7; 12	inteiro
ano	1992; 2008	inteiro
salário	1320,00	real
sexo	'M' ou 'F'	
nome	"Maria"; "João"	
É maior de idade	Verdadeiro ou Falso	

- Quais são os melhores tipos para estas variáveis?

Programas de computador

- **Exemplo:**

- Considere que queremos fazer um programa que vai utilizar variáveis para guardar as seguintes informações:

Tipos possíveis:

caracter cadeia inteiro
real logico

Variável	Valores possíveis	Melhor tipo
dia	1; 3; 31	inteiro
mês	3; 7; 12	inteiro
ano	1992; 2008	inteiro
salário	1320,00	real
sexo	'M' ou 'F'	caracter
nome	"Maria"; "João"	
É maior de idade	Verdadeiro ou Falso	

- Quais são os melhores tipos para estas variáveis?

Programas de computador

- **Exemplo:**

- Considere que queremos fazer um programa que vai utilizar variáveis para guardar as seguintes informações:

Tipos possíveis:

caracter cadeia inteiro
real logico

Variável	Valores possíveis	Melhor tipo
dia	1; 3; 31	inteiro
mês	3; 7; 12	inteiro
ano	1992; 2008	inteiro
salário	1320,00	real
sexo	'M' ou 'F'	caracter
nome	"Maria"; "João"	cadeia
É maior de idade	Verdadeiro ou Falso	

- Quais são os melhores tipos para estas variáveis?

Programas de computador

- **Exemplo:**

- Considere que queremos fazer um programa que vai utilizar variáveis para guardar as seguintes informações:

Tipos possíveis:

caracter cadeia inteiro
real logico

Variável	Valores possíveis	Melhor tipo
dia	1; 3; 31	inteiro
mês	3; 7; 12	inteiro
ano	1992; 2008	inteiro
salário	1320,00	real
sexo	'M' ou 'F'	caracter
nome	"Maria"; "João"	cadeia
É maior de idade	Verdadeiro ou Falso	logico

- Quais são os melhores tipos para estas variáveis?

Programas de computador

- **Exemplo:**
 - Podemos, então, declarar **diversas variáveis** de **diversos tipos** em um programa:

```
programa{  
    funcao inicio(){  
        inteiro dia, mes, ano  
        real salario  
        caracter sexo  
        cadeia nome  
        logico maiorDeIdade  
    }  
}
```

Memória		
Posição	Identificador	Valor
1	dia	?
2	mes	?
3	ano	?
4	salario	?
5	sexo	?
6	nome	?
7	maiorDeIdade	?
...		

Programas de computador

- **Atribuição**

- Para inserir um **valor** em uma variável, precisamos utilizar o **operador de atribuição** “=”.
- Assim, para **guardar um valor** em uma **variável**, basta fazer algo como:

cadeia nome = “Marco”

- Lembrando sempre de usar **tipos compatíveis**.
- Se quisermos **imprimir na tela** o nome salvo, como fazer?

Programas de computador

- **Atribuição**

- Para inserir um **valor** em uma variável, precisamos utilizar o **operador de atribuição** “=”.
- Assim, para **guardar um valor** em uma **variável**, basta fazer algo como:

cadeia nome = “Marco”

- Lembrando sempre de usar **tipos compatíveis**.
- Se quisermos **imprimir na tela** o nome salvo, como fazer?

escreva(nome)

Programas de computador

- **Constantes**

- Algumas vezes, podemos usar um **espaço de memória** para guardar um **valor fixo**, que não muda.
- Se quiséssemos **salvar** o cpf de uma pessoa, por exemplo.
- Neste caso, estaríamos lidando com uma **constante**.
- Uma **constante** tem todas as características de uma **variável**, com a exceção de **não podermos alterar o valor** de uma constante ao longo da execução do programa.

Programas de computador

- Constantes

- A **sintaxe** para declarar uma constante é:

Valor compatível com o tipo declarado

const **tipo** **identificador** = **valor**

Palavra reservada indicando que se trata de uma constante.

Qualquer identificador seguindo as **3 restrições**.

Um dos tipos possíveis:

caracter **cadeia** **inteiro**
real **logico**

Programas de computador

- **Entrada e saída de dados**

- Já vimos como fazer a **saída de dados** de um programa usando a função “escreva”.
- Podemos fazer algo similar para fazer a **entrada de dados**: usar a função “leia”.

leia(**variável**)

- Quando a execução chegar nesta linha, o **console** irá esperar a **digitação** de dados.
- Estes dados serão **salvos** na variável passada como **argumento**.

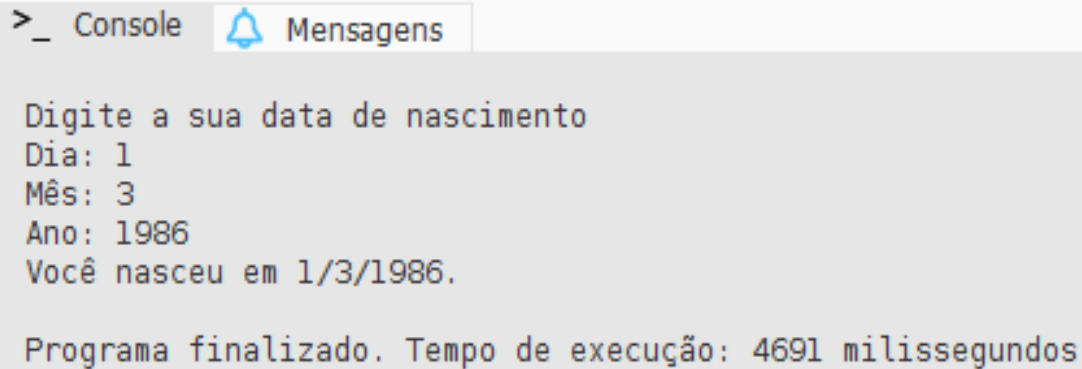
Programas de computador

- **Exemplo**

- Escreva o seguinte programa, analise-o e verifique a saída após execução:

```
programa{  
  
    funcao inicio(){  
        inteiro dia, mes, ano  
        escreva("Digite a sua data de nascimento\n")  
        escreva("Dia: ")  
        leia(dia)  
        escreva("Mês: ")  
        leia(mes)  
        escreva("Ano: ")  
        leia(ano)  
        escreva("Você nasceu em ", dia, "/", mes, "/", ano, ".\n")  
    }  
}
```


Programa



```
> _ Console Mensagens
Digite a sua data de nascimento
Dia: 1
Mês: 3
Ano: 1986
Você nasceu em 1/3/1986.

Programa finalizado. Tempo de execução: 4691 milissegundos
```

- **Exemplo**

- Escreva o seguinte programa, analise-o e verifique a saída após execução:

```
programa{

    funcao inicio(){
        inteiro dia, mes, ano
        escreva("Digite a sua data de nascimento\n")
        escreva("Dia: ")
        leia(dia)
        escreva("Mês: ")
        leia(mes)
        escreva("Ano: ")
        leia(ano)
        escreva("Você nasceu em ", dia, "/", mes, "/", ano, ".\n")
    }
}
```

Programas de computador

- **Comentários**

- São trechos de código que serão **desconsiderados** pelo compilador.

```
programa {  
  
    funcao inicio(){  
        inteiro i  
        escreva("Digite um número inteiro: ")  
        leia(i) // E se o usuário fornecer um número real?  
        /*  
            Se o usuário fornecer um número real, ocorrerá um erro  
            de execução e o programa será abortado.  
        */  
        escreva("Você digitou ", i, ".\n")  
    }  
}
```

Programas de computador

- **Exercícios**

2. Escreva um programa que cria uma variável para cada informação do aluno, lê os valores digitados pelo usuário e imprime na tela conforme mostra a figura abaixo.

----- Informações do aluno -----

Nome:

Idade:

E-mail:

Curso:

Matricula:

Disciplina:

Programas de computador

• Exercícios

3. Escreva um programa para ler:

- O nome (com o sobrenome) de uma pessoa
- Sua data de nascimento (constituída de dia, mês e ano)
- Seu peso (em Kg)
- Sua altura (em metros)
- Seu sexo ('M' ou 'F')

O programa deve, após a leitura destes dados, escrevê-los na tela do computador.

Programas de computador

- **Exercícios**

4. Escreva um programa para ler o nome e o sobrenome de uma pessoa e escrevê-los na seguinte forma: sobrenome seguido por uma vírgula e pelo nome.



Exemplo:

entrada: "Marco"

"Kappel"

saída: "Kappel, Marco"

• Exercícios

5. Quais dos nomes abaixo são válidos para um identificador?

a) A1BC

c) NomeDoAluno

e) funcao

b) XA,1d

d) 198_Aberto

f) média_final

6. Identifique os tipos dos seguintes dados:

a) 0.4

c) 208

e) “informática”

b) ‘ç’

d) ‘8’

7. Qual o tipo de dado ideal para representar os seguintes dados?

a) O conceito de um aluno (A, B, C ou D).

b) A altura de uma pessoa em metros.

c) O número de filhos de uma pessoa.

d) O CEP de uma residência (ex: 25.963-090)

Programas de computador

FIM