

# **Apresentação e Introdução**

## Ferramentas e Aplicações em Biotecnologia

Vasco Ferrinho Lopes  
vasco.lopes@ubi.pt

UBI  
Ano letivo 2023-2024



# Programa e Critérios de Avaliação

- “Ferramentas computacionais:
  - Introdução à programação: variáveis, atribuições, tipos, operadores aritméticos e relacionais.
  - Vetores e matrizes.
  - Programas e sub-rotinas.
  - Instruções de seleção.
  - Ciclos e vectorização.
  - Manipulação de strings.
  - Estruturas de dados.
  - Ficheiros.
  - Criação de gráficos.
  - Utilização de bases de dados biológicas.”
- **1 frequência prática** a ser feita no dia 2-11-2013 (a confirmar se pode alterar para a última aula prática, no dia 27-10)



# O Que é Programação?

- **Programação** é o ato de escrever instruções e algoritmos com o fim de resolver problemas ou realizar tarefas com meio de um computador.
- **Programar** é o ato de **criar um programa**, escrevendo-o numa determinada **linguagem de programação**



“The computer **programmer** is a **creator of universes** for which **he** alone is the **lawgiver.** - Joseph Weizenbaum, 1976



# Motivação

## Porque aprender a programar?

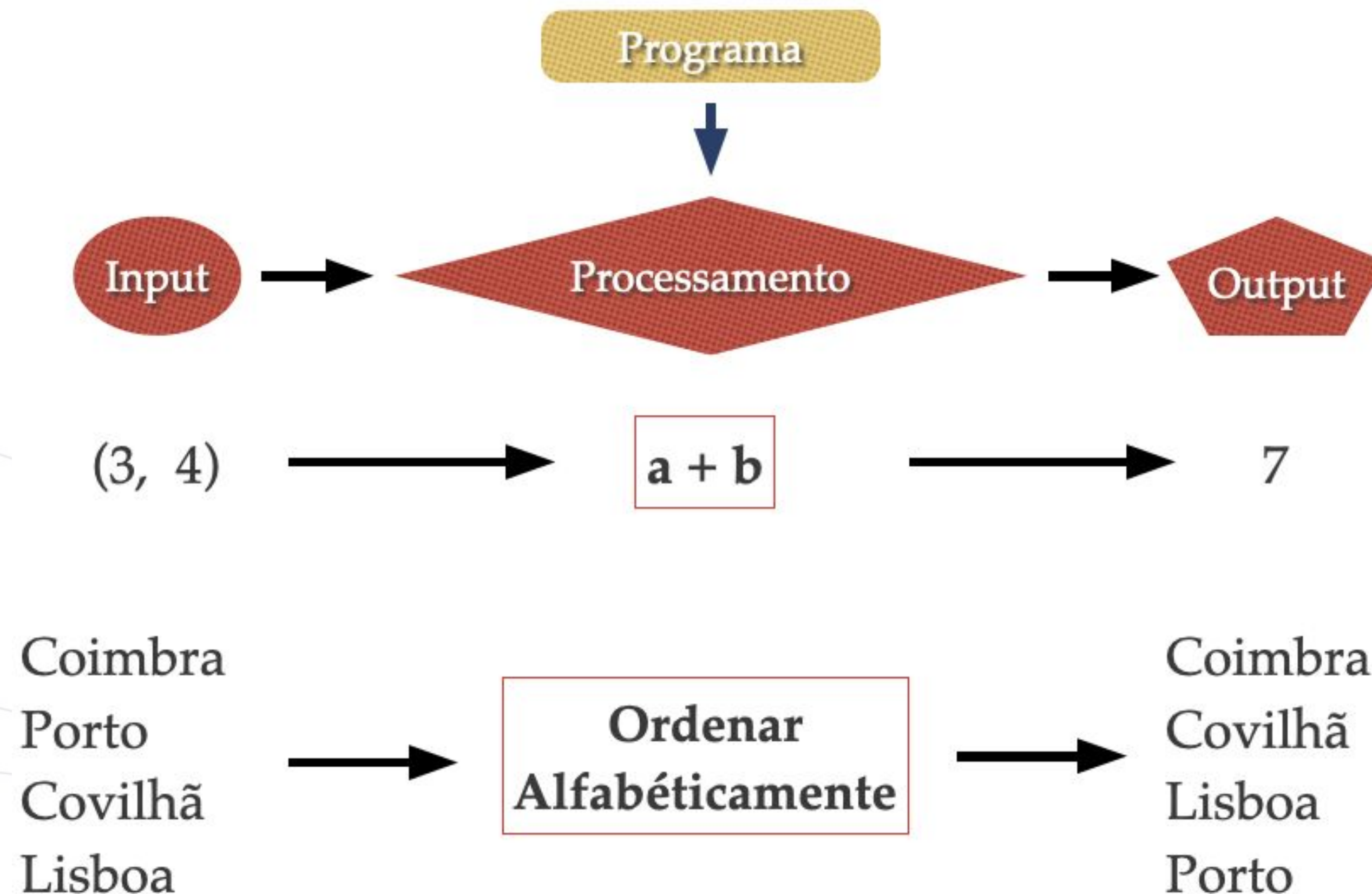
- Programar permite **criar soluções** para **problemas complexos** com apoio de máquinas
- Desenvolver **pensamento lógico e computacional**. Aumenta a nossa capacidade de **resolução de problemas** e de **análise lógica**
- Melhorar a **capacidade analítica**. Desenhar programas requer atenção aos detalhes, raciocínio lógico e percepção/previsão de potenciais problemas
- Aplicações ilimitadas:
  - Desenvolver e criar aquilo que imaginamos
  - Transferência de conhecimento
  - Empregabilidade





# Motivação

Um **programa** é um **conjunto de instruções** que **controlam** uma **máquina**. Esse controlo pode envolver a leitura e armazenamento de dados, bem como a transformação destes em informação subsequente.





# Motivação

Um **programa** é então um **conjunto de instruções** de controlo para uma máquina, expressas numa **linguagem rigorosa e precisa**. O programador **codifica** as instruções nessa linguagem, e a máquina “**interpreta**” e “**executa**” essas instruções.

- **Linguagem Natural:** ambígua e difícil de codificar totalmente. Ex: “A UBI é na Covilhã”
- **Linguagem Máquina:** “01010101 01000010 01001001” rigorosa mas difícil para humanos
- **Linguagem de Alto Nível:** ponto intermédio entre as duas anteriores.  
se valor < 0 então: “valor inválido”.
- Existem também **linguagens de baixo nível** que se aproximam mais de código máquina (ex: assembly).



# Python

**Python** é uma **linguagem de programação interpretada** de **alto nível** com uma **sintaxe simples**, o que a torna facilmente legível e extremamente amigável. Originalmente construído para satisfazer o desejo de Guido Van Rossum por uma linguagem de programação simples de usar e bonita de se ver, Python foi lançado ao mundo pela primeira vez em 1991.

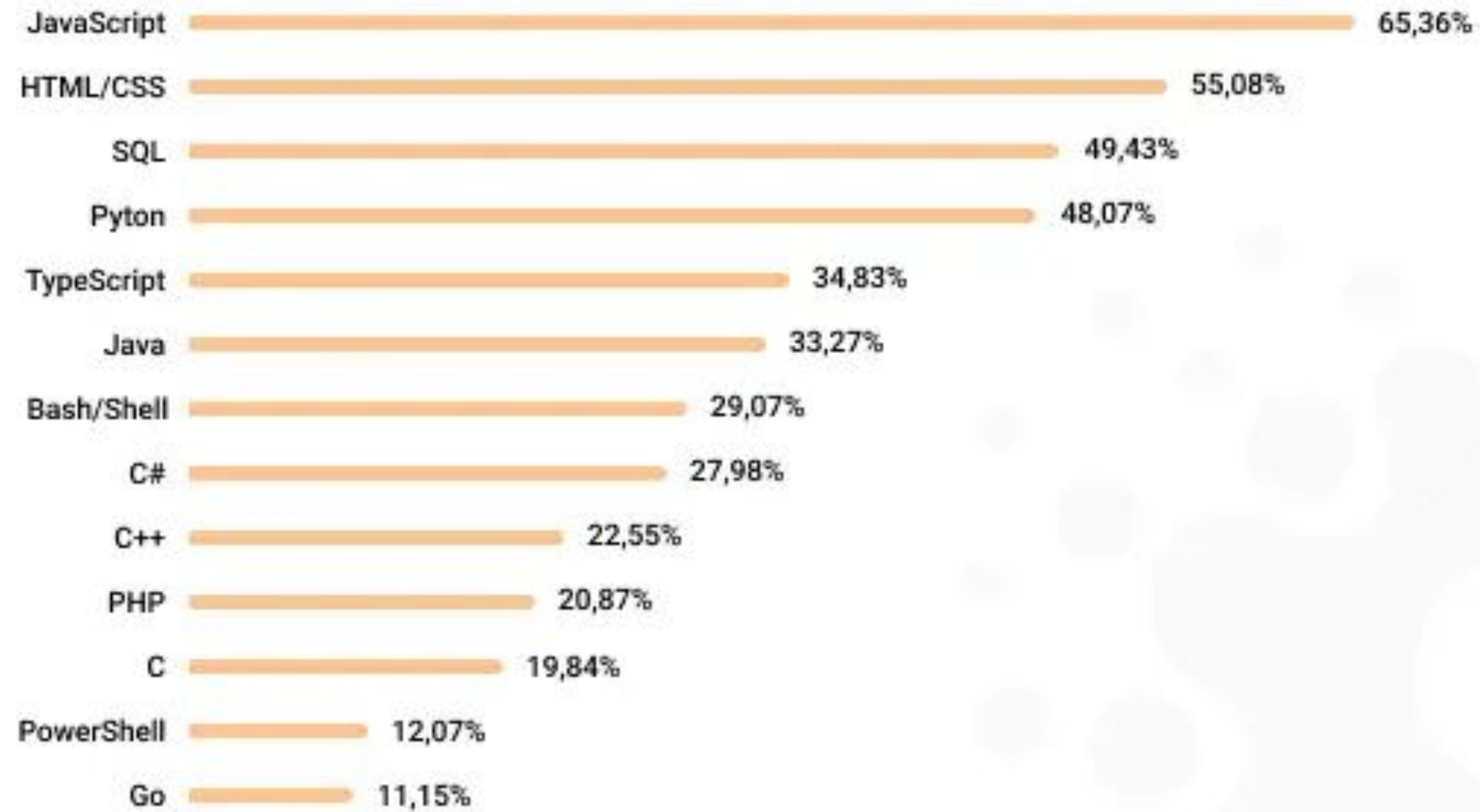
- De **fácil leitura**: usa palavras-chave em inglês em vez de pontuação. As quebras de linha ajudam a definir os blocos de código. Significando que se pode identificar a função do código simplesmente olhando para ele.
- **Open-source**: o código-fonte é totalmente público e qualquer pessoa pode propor mudanças.
- **Portável**: trabalha em múltiplas plataformas sem necessidade de alterações.
- **Amplas bibliotecas**: qualquer utilizador pode utilizar funções pré-definidas.





# Python

## The most widely used programming languages worldwide







# Python

Em qualquer linguagem existe um vocabulário, incluindo pontuação e regras. Em python, não é diferente:

- Palavras reservadas (**keywords**)
- Identificadores (**identifiers**)
- Variáveis (**variables**)
- Operadores (**operators**)
- Separadores (**punctuators**)

and	as	assert	async	await
break	class	continue	def	del
elif	else	except	False	finally
for	from	global	if	import
in	is	lambda	None	nonlocal
not	or	pass	raise	return
True	try	while	with	yield



# Python

Uma **variável** é uma entidade de um programa que **pode assumir valores** durante o seu tempo de existência em um programa

- Uma variável é representada por um identificador e corresponde a uma porção na memória principal do computador
- O operador de atribuição (“=”) é o que permite associar um valor a uma variável. Também se diz guardar ou armazenar um valor na variável

`a = 0`

`x = 37.5`

`b = 1`

`y = -1.9`

`c = True`

`d = False`

`curso = “Biotecnologia”`



**Operadores** permitem arranjar e compor outros elementos em expressões mais complexas, segundo certas regras sintáticas.

Categoria	Operador
Aritméticos	+ - * / % ** //
Atribuição	= += -= *= /=
Comparação	== != > < >= <=
Lógicos	and or not
Identidade	is is not
Membrasia	in not in
Bitwise	&   ^ « »



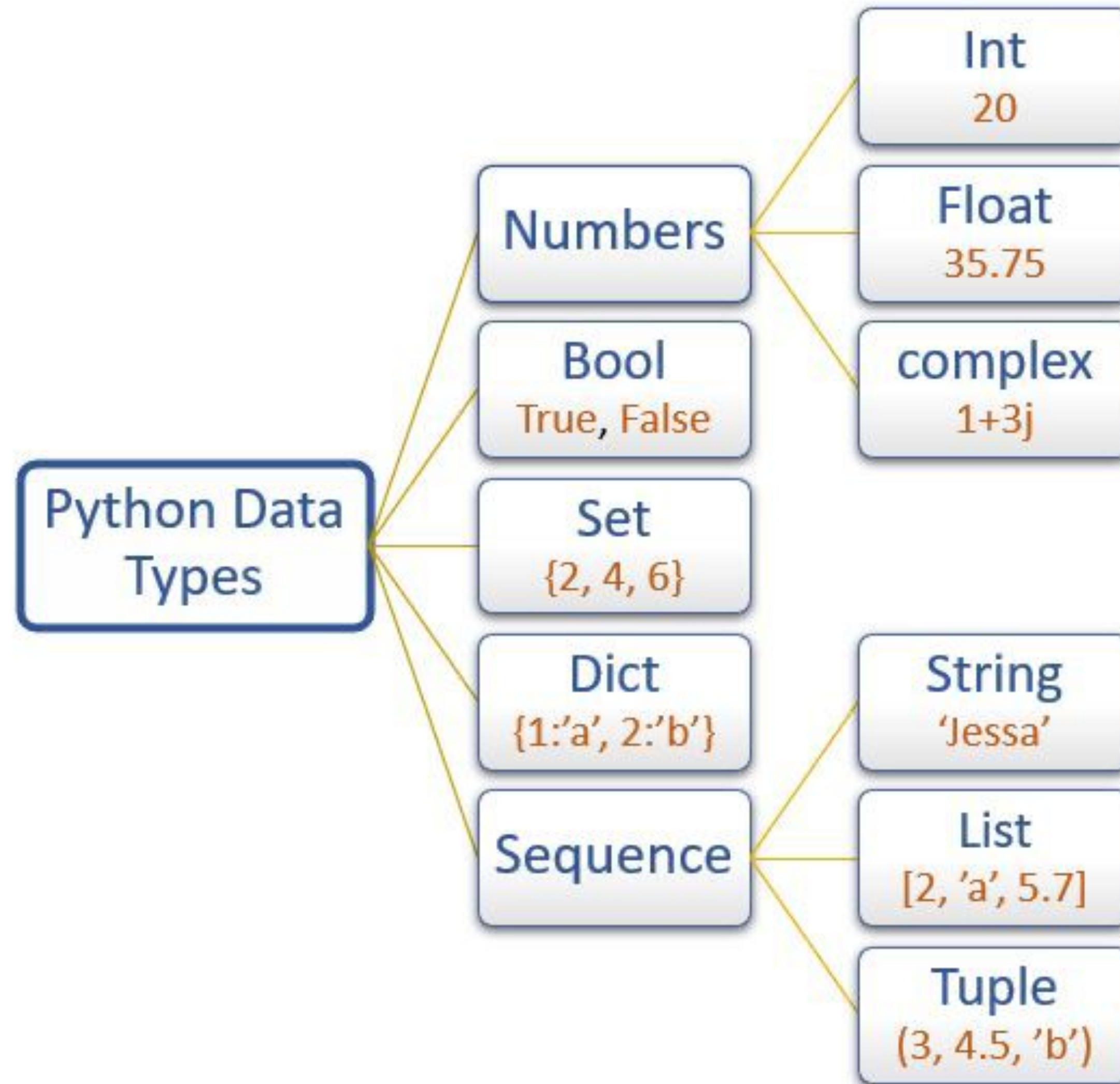
**Separadores** são símbolos gráficos ou combinações de símbolos gráficos que são usados na construção de expressões e delimitação de instruções.

( ) [ ] { } , ;





# Python





# Python

Variáveis de um tipo de dados, também podem vir a ser convertidas em outros tipos de dados

Origem	Destino	Função	Exemplo
string	integer	<code>int(s:str) → int</code>	<code>int('23')</code>
string	float	<code>float(s:str) → float</code>	<code>float('23.7')</code>
integer	string	<code>str(a:int) → str</code>	<code>str(78)</code>
float	string	<code>str(x:float) → str</code>	<code>str(7.87)</code>
char	integer	<code>ord(c:char) → int</code>	<code>ord('e')</code>
integer	char	<code>chr(k:int) → char</code>	<code>chr(97)</code>

```
int('23'), float('23.7'), str(78), chr(97), ord('e')
(23, 23.7, '78', 'a', 101)
```



# Edição e Execução de um Programa

Durante as nossas aulas vamos trabalhar com o “Google Colaboratory” ou o Visual Studio Code. Para o Google Colab podemos usar a google drive a partir de uma conta gmail.

The screenshot displays the Google Colaboratory web interface. At the top, the notebook is titled 'Untitled0.ipynb'. The menu bar includes 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help', with a status indicator 'All changes saved'. On the right, there are links for 'Comment', 'Share', and a user profile icon. Below the menu, a toolbar shows '+ Code' and '+ Text' options. The main area contains a code cell with the following Python code:

```
#  
# Primeiro Programa  
#  
print("Olá Mundo")
```

Below the code cell, the output is displayed as 'Olá Mundo'. The bottom of the interface shows a terminal area with a play button icon and a vertical scrollbar. On the far right, there are indicators for 'RAM' and 'Disk' usage, both showing green bars and a checkmark.





```
print("Hello Humans")
```

Hello Humans

```
a = 1
```

```
b = 5
```

```
print(a+b)
```

6

```
if (a > b):
```

```
    print ("A é maior que B")
```

```
elif (b > a):
```

```
    print ("B é maior que A")
```

```
else:
```

```
    print ("São iguais")
```

B é maior que A





```
# Programa que converte temperatura em graus Fahrenheit para graus Celsius
F = int(input("Qual o valor em Fahrenheit? "))
C = 5/9 * (F-32)
print(C)
```

```
Qual o valor em Fahrenheit? 5
-15.0
```

# Bibliografia:

Costa, Ernesto 2015. Programação em Python, Fundamentos e Resolução de Problemas.  
Editora FCA.

