

python™

# Aula 01 - Introdução a Programação Estruturada

UNIP - CIÊNCIA DA COMPUTAÇÃO

DISCIPLINA: IPE

PROFESSORES: CÉLIO / LUCIANA



# LINGUAGEM PYTHON

---

- Python foi lançada por Guido van Rossum em 1991.
- É uma Linguagem de Programação de alto nível, interpretada e orientada a objetos.
- Programas como Dropbox, Reddit e Instagram são escritos em Python.
- Python também é a linguagem mais popular para análise de dados e conquistou a comunidade científica.



# LINGUAGEM PYTHON

---

- Projetada para ser simples e ao mesmo tempo eficaz.
- Pode ser utilizada para a criação de programas diversos:
  - processadores de texto;
  - planilhas eletrônicas;
  - tarefas de sistemas operacionais;
  - programas de comunicação;
  - programas para a automação industrial;
  - programas para a solução de problemas da Engenharia, Física, Química e outras Ciências...

# Para ser um bom programador:

---

**1º passo:** entender o problema a ser resolvido;

**2º passo:** Com base no 1º passo, estruturar um algoritmo, entendendo:

**ENTRADA** → **PROCESSAMENTO** → **SAÍDA** ou **RESULTADO**

**3º passo:** Digitar o código fonte na linguagem desejada (usando um ambiente de desenvolvimento)

*\*\*\* Usaremos o Python no Visual Studio Code*

**4º passo:** Analisar as mensagens de erros e corrigi-los até ficar correto (zero erros);

**5º passo:** Verificar se o programa satisfaz o objetivo.

# Instalação e configuração do ambiente

---

Acesse o link abaixo para fazer a instalação do Python e do Visual Code:

<https://docs.microsoft.com/pt-br/learn/modules/python-install-vscode/>

Opção de compilador online:

<https://www.online-python.com/>

# Exemplo de Problema a ser resolvido:

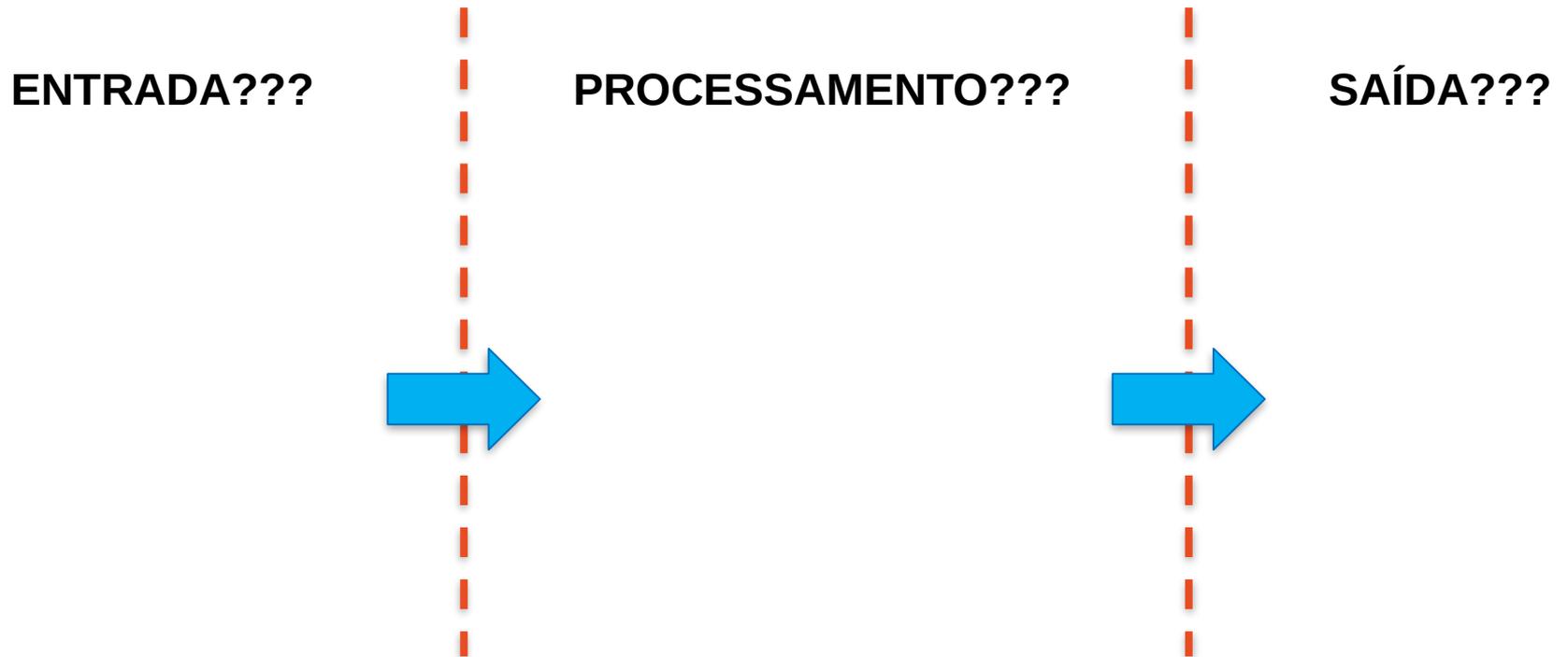
---

Permita a digitação de um número inteiro qualquer, calcule e imprima na tela o dobro desse número.

# Exemplo de Problema a ser resolvido:

---

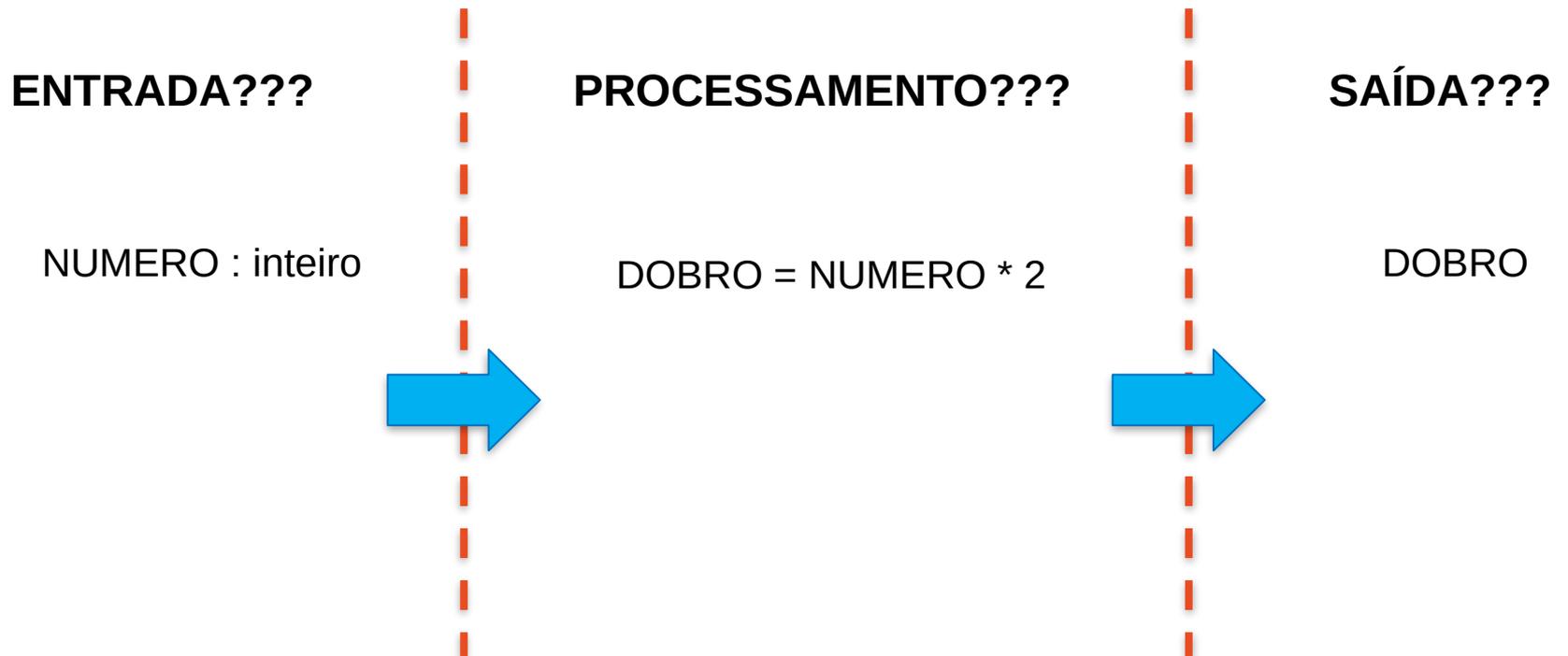
Permita a digitação de um número inteiro qualquer, calcule e imprima na tela o dobro desse número.



# Exemplo de Problema a ser resolvido:

---

Permita a digitação de um número inteiro qualquer, calcule e imprima na tela o dobro desse número.



# Algoritmo (portugol)

---

Definiremos um algoritmo para o problema anterior, que poderá depois ser escrito em qualquer linguagem:

```
ALGORITMO DOBRO_NUM;  
DECLARE  
    NUM:INTEIRO;  
    DOBRO:INTEIRO;  
INICIO  
    ESCREVA ( “ Digite um Número Inteiro” );  
    LER( NUM );  
  
    DOBRO <- NUM * 2;  
  
    ESCREVA ( “ O dobro do número digitado é =” , DOBRO);  
FIM
```

# Programa Python

---

Escrevendo o algoritmo na linguagem Python:

```
num = int(input('Digite um Número Inteiro '))  
dobro = num * 2  
print('O dobro do número digitado é = ', dobro)
```

**Abra o Visual Studio e Bora colocar a**



# Comparação Portugol Vs. Código Python

---

```
ALGORITMO DOBRO_NUM;  
DECLARE  
    NUM:INTEIRO;  
    DOBRO:INTEIRO;  
INICIO  
    ESCREVA ( " Digite um Número Inteiro" );  
    LER( NUM );  
    DOBRO <- NUM * 2;  
    ESCREVA ( " O dobro do número digitado é =" , DOBRO);  
FIM
```

```
num = int(input('Digite um Número Inteiro '))  
dobro = num * 2;  
print('O dobro do número digitado é = ', dobro);
```

# ATENÇÃO:

## **Não se precipite !!!**

---

- O Portugol é muito importante! Com ele focamos na lógica e ignoramos as “burocracias” que algumas linguagens possuem;
- Não se assustem, os códigos em Python normalmente são menores mesmo (até comparado com outras linguagens);
- Quando houver dificuldades com o Python, idealize o algoritmo primeiro no Portugol e se precisar construa o quadro de “Entrada, Processamento e Saída”.

# LINGUAGEM INTERPRETADA X COMPILADA

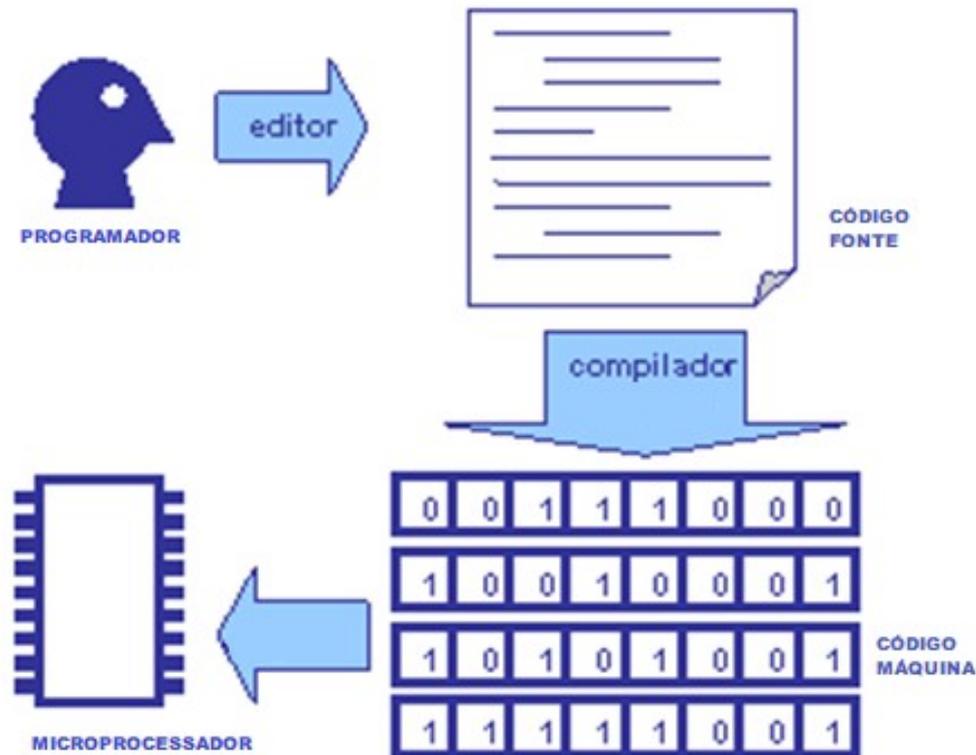
---

As linguagens de programação podem ser **compiladas** ou **interpretadas**. Iremos analisar essas duas abordagens:

**Compilação** é o processo de transformar o código fonte, escrito em uma linguagem de alto nível, em um código binário específico para a arquitetura do computador (ou máquina virtual) que está sendo utilizado.

# LINGUAGEM INTERPRETADA X COMPILADA

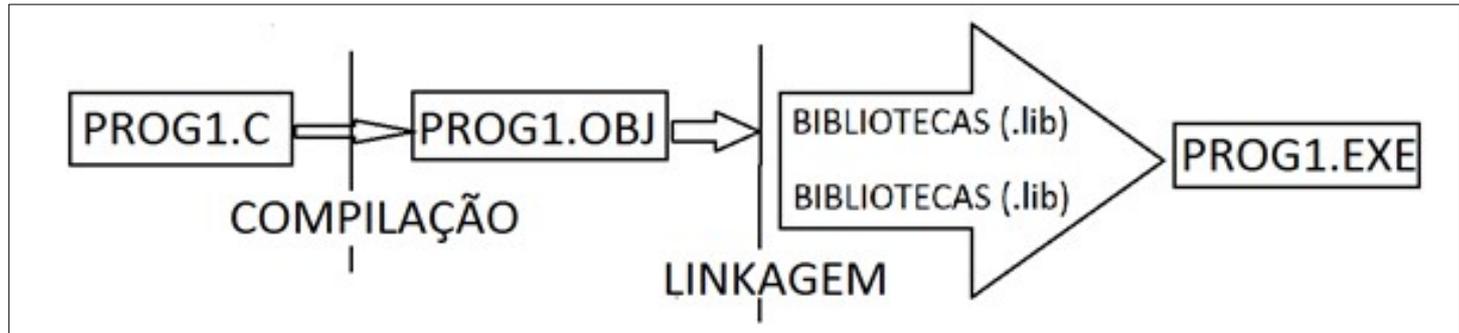
## Compilação



# LINGUAGEM INTERPRETADA X COMPILADA

---

## *Compilação (exemplo linguagem C)*



# LINGUAGEM INTERPRETADA X COMPILADA

---

**Interpretação** ocorre quando o código é executado diretamente do arquivo fonte. O interpretador (assim chamado) lê cada instrução do código-fonte em alto nível, realiza a “tradução” para linguagem de máquina e o executa no mesmo instante, sem a necessidade de uma fase de compilação.

```
num = int(input('Digite um Número Inteiro '))
dobro = num * 2;
print('O dobro do número digitado é = ', dobro);
```

*Interpreta e executa nesses passos:*

```
1 num = input('Digite um Número Inteiro ')
2 res = int(num)
3 dobro = res * 2
4 print('O dobro de ', num, ' é: ', dobro)
```

# Exercícios:

---

## **1 - Missionários e canibais:**

**Objetivo:** Levar todos de uma margem para a outra

**Regras:** Os missionários não podem ficar em menor número, em qualquer uma das margens, ou os canibais os devorarão.



**[Anote sua sequencia no caderno e depois teste se deu certo aqui:](https://rachacuca.com.br/jogos/missionarios-e-canibais/)**

**<https://rachacuca.com.br/jogos/missionarios-e-canibais/>**

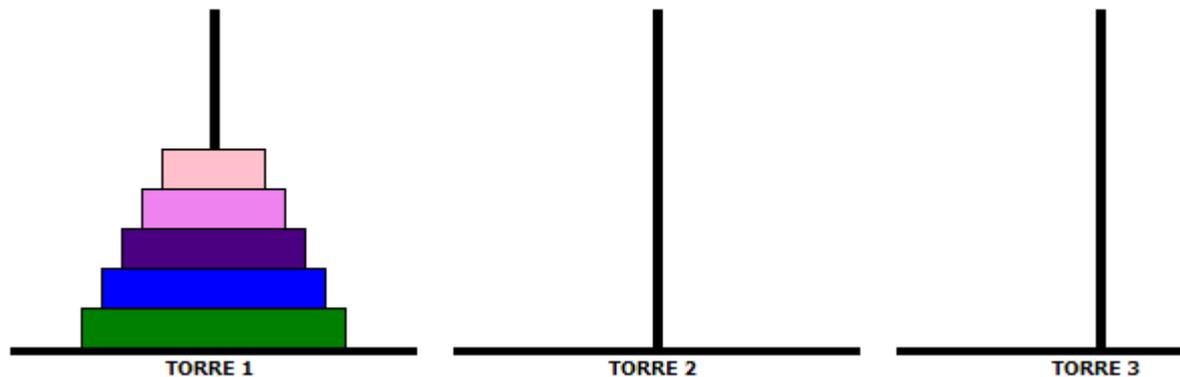
# Exercícios:

---

## 2 - Torre de Hanói

**Objetivo:** Mover todos os discos para o pino da direita

**Desafio:** Tentar com 3, 4 e 5 discos



[Anote sua sequencia no caderno e depois teste se deu certo aqui:](https://www.somatematica.com.br/jogos/hanoi/)  
<https://www.somatematica.com.br/jogos/hanoi/>

# Exercícios:

---

## 3 – O leão, a zebra e o feno

**Descrição:** Escreva um algoritmo para levar um leão, uma zebra e um feno de um lado para outro de um rio, com um bote (1 ou 2 passageiros). Sabe-se que o leão não pode ficar sozinho com a zebra e nem a zebra sozinha com o feno.

