



Projeto de Pesquisa

Orientador

Prof. Golbery Aguiar

Coorientadores

Prof. Danyllo Wagner

Prof.^a Ianna Sodre

Pesquisadores

Gabriel William

Jhonnata Virginio

João Gabriel

Luiz Eduardo



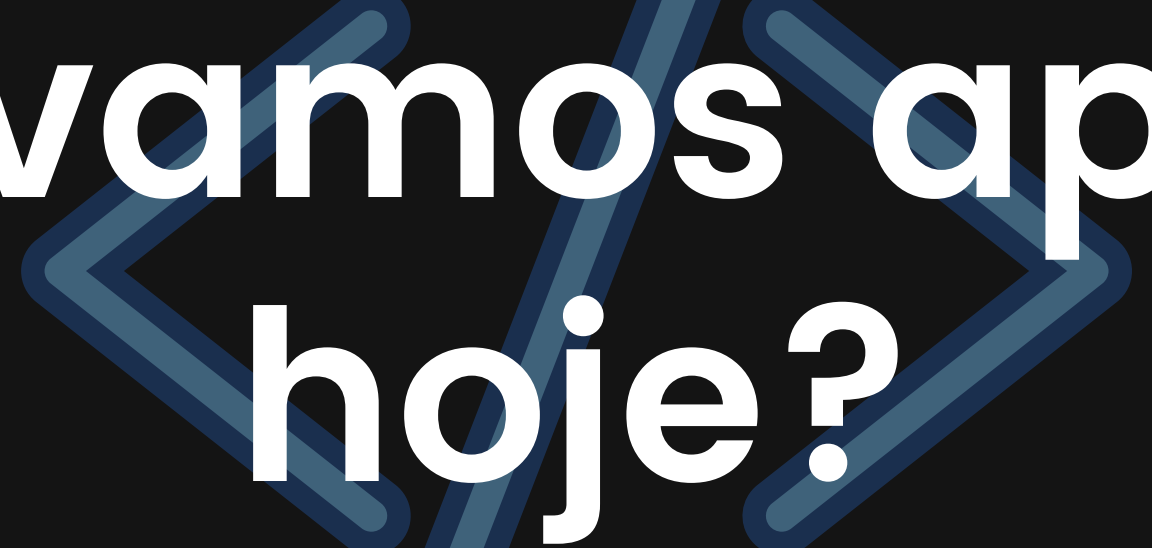
LAPLIN
LABORATÓRIO DE ANÁLISE E
PROCESSAMENTO DE LINGUAGEM NATURAL



**INSTITUTO
FEDERAL**
Paraíba

Campus
Campina Grande

O que vamos aprender
hoje?

A decorative graphic consisting of several overlapping, semi-transparent blue lines that form a complex, geometric shape resembling a stylized diamond or a series of nested triangles. The lines are thick and have a slight gradient, giving it a three-dimensional appearance.

CONTEÚDOS

- **Introdução à algoritmos**
- **Entrada e saída de dados**
- **Variáveis**
- **Tipos de variáveis**

Algoritmo

“Um algoritmo é uma sequência de instruções bem definidas para cumprir uma tarefa”

- Um algoritmo pode ser comparado à uma receita de bolo. Uma série de passos bem definidas que ao final cumprem a tarefa de cozinhar um bolo.



Exemplo:

Um algoritmo para cozinhar um bolo seria:

1. Reunir os ovos
2. Reunir a farinha
3. Reunir a manteiga
4. Reunir o leite
5. Pegar a forma

1. Quebrar os ovos
2. Adicionar a farinha
3. Adicionar o leite
4. Adicionar manteiga
5. Misturar até homogeneizar

1. Despejar na forma
2. Pré-aquecer o forno
3. Inserir a forma no forno
4. Esperar 30 min
5. Retirar o bolo do forno

Exemplo:

Um exemplo de pseudocódigo seria:

```
ingredientes = ["ovos", "farinha", "leite", "manteiga"]
```

```
ingredientes_prontos = preparar_ingredientes(ingredientes)
```

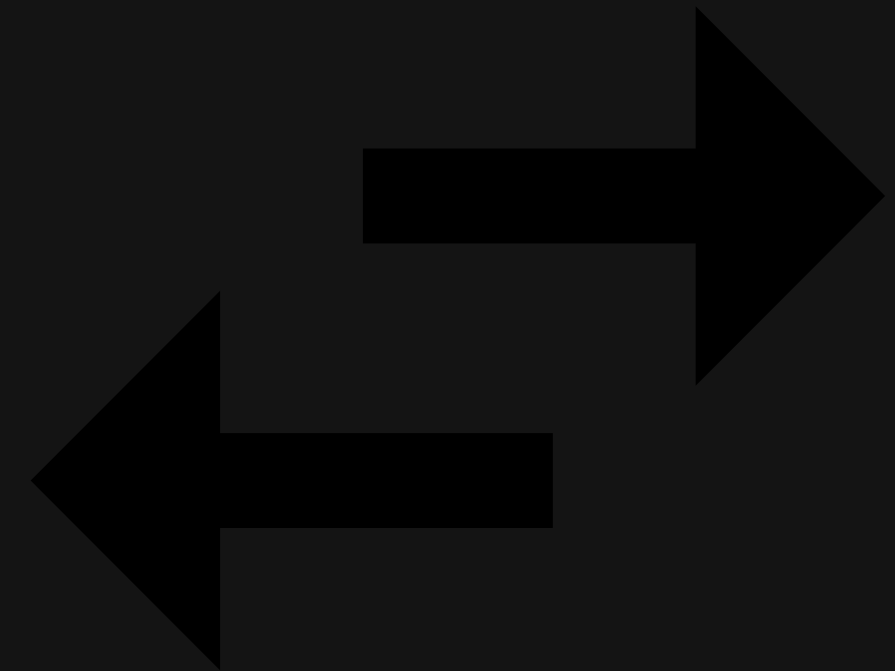
```
forma = misturar(ingredientes_prontos)
```

```
bolo = forno(forma)
```

Entrada e saída de dados

Antes de tudo, precisamos aprender a nos comunicar com o computador.

Para isso precisamos aprender a utilizar comandos de fluxo de dados.



```
print()  
input()
```


Saída de dados

- O comando **print** é responsável pela saída dos dados

Comando

```
print("Olá mundo")
```

Resultado

Olá mundo

Nosso primeiro programa

Escreva um programa que exiba na tela
“Olá mundo, meu nome é (SEU NOME)”

```
print("Olá mundo (seu nome aqui)")
```

Como faríamos o computador perguntar o nosso nome?

Para isso, utilizamos comandos de entrada de dados!

- O comando para realizar entrada de dados é o **input**

```
input("insira o seu nome:")
```

Como isso funciona?

Vamos supor que a resposta foi **João**

O computador substitui o termo **input()** pela sua resposta

```
input("insira o seu nome:")
```



```
"João"
```

Sendo assim

Uma das formas de exibir o seu nome na tela,
após o computador perguntar, seria:

```
print(input("insira o seu nome:"))
```

Como isso funciona?

```
print(input("insira o seu nome:"))
```

Vamos dividir o nosso algoritmo em partes.
Supondo que a resposta foi **João**

Primeiro o programa
pergunta o seu nome
no terminal.

```
input("insira o seu nome:")
```

"João"

depois substitui
o termo input()
pelo seu nome

Como funciona?

Depois da substituição, voltamos ao que já conhecemos.

O comando print irá exibir o nome **João** que foi inserido.

```
print(input("insira o seu nome:"))
```

```
input("insira o seu nome: ")
```

```
"João"
```

```
print("João")
```

Variáveis

Imagine agora que você precisasse utilizar o seu nome para exibir na tela outra vez, ou precisasse guardar este valor para usar novamente.

Este é o conceito de variável, uma pequena unidade de armazenamento de memória que guarda dados.

```
nome = input("qual o seu nome? ")
```


Como funciona?

```
nome = input("qual o seu nome? ")
```



```
print(nome)
```

Tipos de variáveis

```
int()
```



Numero inteiro

```
float()
```



Numero racional

```
str()
```



Texto

```
bool()
```



Verdadeiro ou falso

Three interlocking gears are visible in the background, rendered in a dark gray color with a slightly lighter gray outline. They are arranged in a triangular pattern, with one gear at the top and two below it, slightly offset to the left and right.

**Agora, vamos
praticar!**



Projeto de Pesquisa

Orientador

Prof. Golbery Aguiar

Coorientadores

Prof. Danyllo Wagner

Prof.^a Ianna Sodre

Pesquisadores

Gabriel William

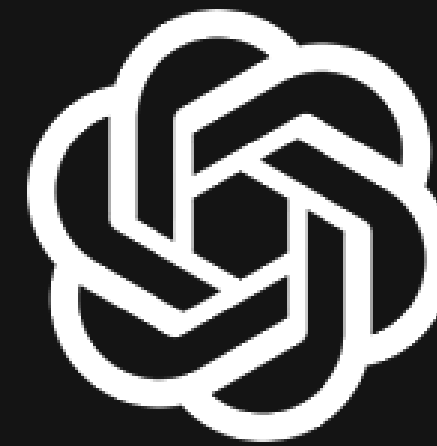
Jhonnata Virginio

João Gabriel

Luiz Eduardo



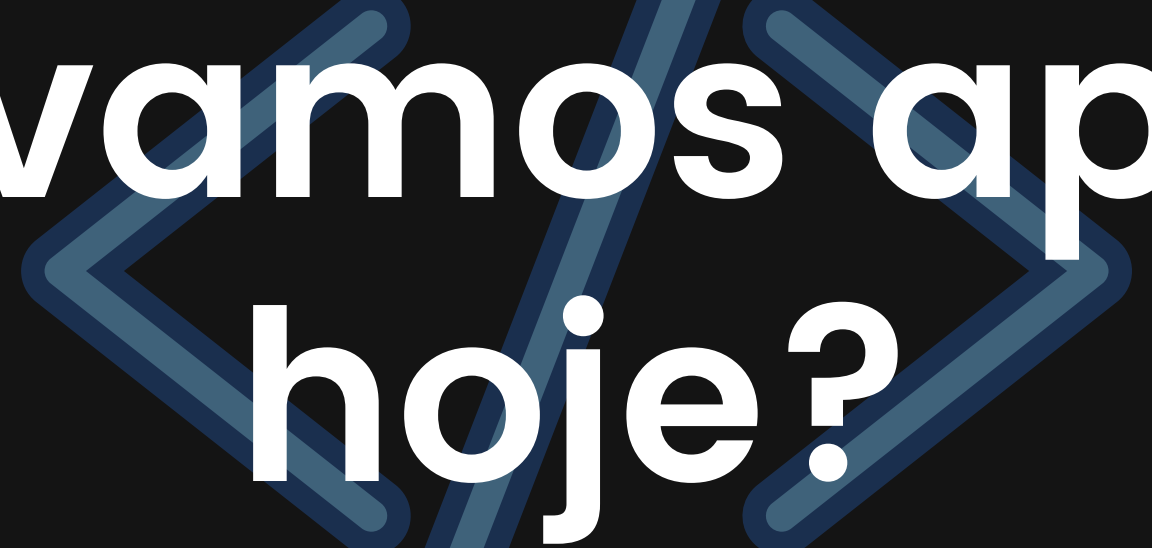
LAPLIN
LABORATÓRIO DE ANÁLISE E
PROCESSAMENTO DE LINGUAGEM NATURAL



**INSTITUTO
FEDERAL**
Paraíba

Campus
Campina Grande

O que vamos aprender
hoje?

A decorative graphic consisting of several overlapping squares with blue outlines, creating a sense of depth and movement. A thick blue diagonal line runs from the top-left towards the bottom-right, passing through the center of the composition.

Estruturas de Decisão ou Condicional



- Operadores aritméticos
- Tipos de Comandos Condicionais
- IF Simples/Composto
- Operadores de Comparação
- Operadores Lógicos
- Aninhamento de IF

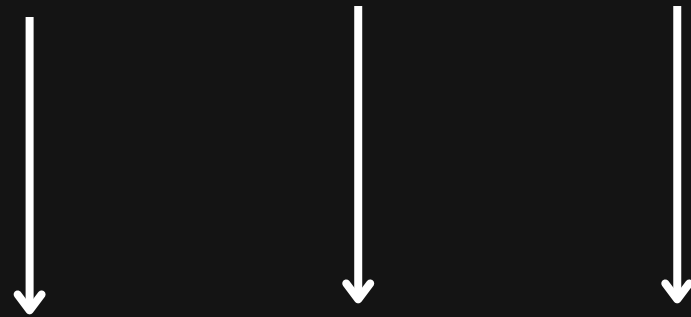
Operadores aritméticos

OPERAÇÃO	OPERADOR
SOMA	+
SUBTRAÇÃO	-
MULTIPLICAÇÃO	*
DIVISÃO	/
DIVISÃO INTEIRA	//
RESTO DE DIVISÃO	%
POTÊNCIA	**

Como funciona?

Para realizar uma operação aritmética o comando é o seguinte:

NÚMERO OPERADOR NÚMERO



1000 + 1000

Como funciona?

Onde o Python encontrar a expressão aritmética ele irá substituir pelo resultado da sua operação.

Por exemplo:

```
print(1000 + 1000)
```



```
print(2000)
```

Ordem de execução

Python por padrão utiliza a ordem de resolução das operações como a da matemática.

Porém, é possível alterar a ordem utilizando parênteses ()

As operações entre parêntesis são prioritárias em relação às demais.

As operações são executadas de baixo para cima nessa tabela

OPERAÇÃO	OPERADOR
SOMA	+
SUBTRAÇÃO	-
MULTIPLICAÇÃO	*
DIVISÃO	/
DIVISÃO INTEIRA	//
RESTO DE DIVISÃO	%
POTÊNCIA	**

Exemplo:

$1 + 1 * 2$

A expressão acima retornará 3, pois a multiplicação entre 1 e 2 acontece primeiro.

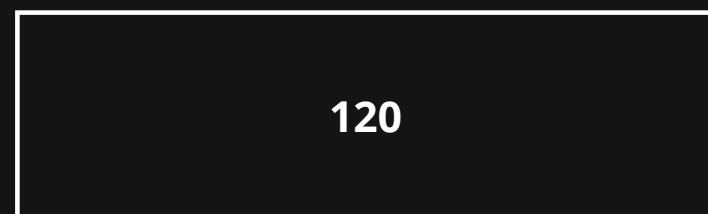
$(1 + 1) * 2$

A expressão acima retornará 4, pois a soma entre 1 e 1 acontece primeiro.

Vamos praticar:

Crie um programa que receba um valor em segundos e transforme-os em minutos.

Entrada



Saída

