

Aula1.py

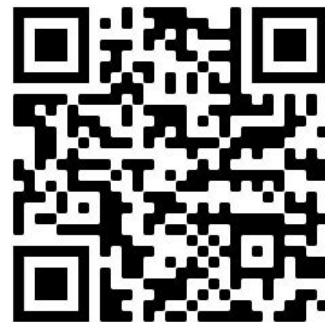
teste\_old.ipynb

```
1
2
3 Programming 'Language' {
4     [Bem Vindo(a)s]
5
6     < Vai dar tudo certo! >
7
8
9
10
11
12 }
13
14
```

Programming Language

# Introdução à Lógica De Programação

- Análise e Desenvolvimento de Sistemas - UNIPÊ / Estacio SA
- MBA Engenharia e Administração de Dados - UNIESP
- Data Intelligence - A3Data
- Co-Founder & CEO - V8tech
- Professor Membro - Artificial Intelligence and Data Analytics - AIDA
- Professor Membro - GDG Paraíba
- Professor - UNIESP



# I.P – 0 que vamos aprender?

- \* Introdução a Lógica de Programação
- \* Algoritmos e Fluxogramas / Tipos de Dados, Variáveis e Constantes
- \* Introdução ao Python
- \* Operadores Aritméticos, Relacionais e Lógicos
- \* Desvio Condicional
- \* Estrutura de Repetição / Web Scraping
- \* Listas, Tuplas e Dicionários
- \* Funções e Subprogramas
- \* Tratamento de Arquivos



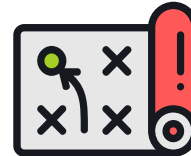
**\* Exercícios**

**\* Avaliações Escritas + (Testes de Mesa)**

**\* Projeto:**

## Lista de Exercicios;

## Calculadora ou Jogo da forca ou outro...



1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14



[www.beecrowd.com.br](http://www.beecrowd.com.br)



Data Science  
Academy

<https://www.datascienceacademy.com.br>

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14



# 01 Logica

< O que é Lógica? >

## 02 Lógica de Programação

< Sim, mais o que tem em  
Lógica de Programação? >

## 03 Algoritmos

< Construir Algoritmos  
Coerentes e válidos >

01 {

[Lógica]

< 0 que é Lógica?



1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14



## 01 {

## [Alguns Entendimentos]

&lt;


- “[a] arte de pensar bem”;
- “[a] ciência das formas do pensamento”;
- “[o] estudo da correção do raciocínio”;
- “[a] ordem da razão;
- “A lógica estuda e ensina a colocar ‘ordem no pensamento’”.

&gt;


}

FORBELLONE, A. L. V. *Lógica de Programação: a construção de algoritmos*. São Paulo: Makron Books, 2005.

## Exemplo += 1 {

```
1  
2  
3 | [  ] < Toda ave é um animal  
4 |                               Todo pássaro é uma ave  
5 |                               Portanto, todo pássaro é um animal  
6 |                               >  
7 | }
```

## Exemplo ## 2 {

```
10 | [  ] < Todo nordestino é brasileiro  
11 |                               João é nordestino  
12 |                               Portanto, João é brasileiro  
13 |                               >  
14 | }
```

## Exemplo += 1 {



```
< Toda ave é um animal  
  Todo pássaro é uma ave  
  Portanto, todo pássaro é um animal  
>
```

}

## Exemplo ## 2 {



```
< Todo nordestino é brasileiro  
  João é nordestino  
  Portanto, João é brasileiro  
>
```

}

```
1  Lógica; {
```

```
2  
3  
4      “Esse é um dos objetivos da lógica, o estudo de  
5      técnicas de formalização, dedução e análise que  
6      permitam verificar a validade de argumentos”  
7  
8
```

```
9  
10  
11  
12      – FORBELLONE, 2005.  
13  
14
```

```
15      ; → Não esqueça do ponto e vírgula
```

```
16  }
```



## Exemplo (1+2) {



A garagem está fechada

0 carro está dentro da garagem

É preciso abrir a garagem para depois sair com o  
carro;

}

## Exemplo -4 {



% Eu leio 5 livros por semana %

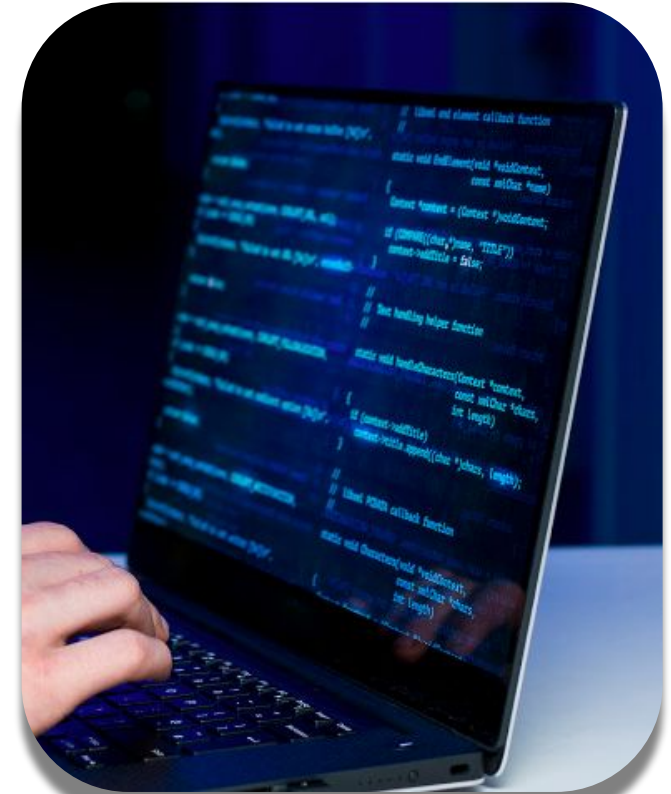
'Minha mãe lê 6 livros a mais do que eu'

(Logo, minha mãe é a que ler mais livros)

}

02 {

# [Lógica de Programação]



# 02 {

- Raciocínio humano é abstrato e intangível;
- Seres humanos expressam o raciocínio de diversas formas:
  - Fazendo uso de várias linguagens e convenções;
- Uma mente treinada pode desenvolver lógica de programação;
- É possível representar o pensamento em linguagens de programação.

}

# Conceito {



- O uso correto das **ordem da razão** → **Lógica**
- Utilizando **simbolização formal da programação** de computadores → **Linguagens (de programação)**
- Que produzam **soluções logicamente válidas e coerentes** → **Programas**
- **Resolvendo com qualidade problemas** → **Resolva os requisitos**







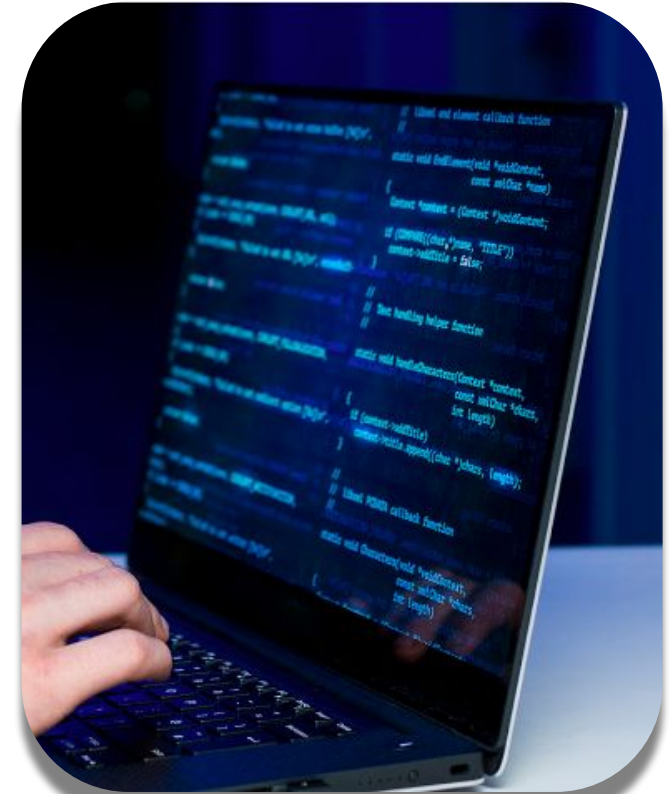
/03

# /ALGORITMOS



02 {

# [Lógica de Programação]



03

{

[Algoritmos]

```
if ($(window).scrollTop() > header1_in
    if (parseInt(header1.css('padding-top',
        header1.css('padding-top', '' +
    } else {
        header1.css('padding-top', '' +
    }

if ($(window).scrollTop() > header2_in
    if (parseInt(header2.css('padding-top',
        header2.css('padding-top', '' +
    } else {
        header2.css('padding-top', '' + header2_in
```

# Algoritmos ◇ {

“O objetivo principal do estudo da lógica de programação é a construção de algoritmos coerentes e válidos”

– **FORBELLONE, 2005.**

() → Quando colocar um parênteses, sempre lembre de fechá-lo.

}



[O que é um  
Algoritmo?]

```
if ($(window).scrollTop() > header1_initialDistance) {  
  if (parseInt(header1.css('padding-top'), 10) > header1_initialPadding) {  
    header1.css('padding-top', '' + $(window).scrollTop() + header1_initialPadding);  
  }  
} else {  
  header1.css('padding-top', '' + header1_initialPadding);  
}  
  
if ($(window).scrollTop() > header2_initialDistance) {  
  if (parseInt(header2.css('padding-top'), 10) > header2_initialPadding) {  
    header2.css('padding-top', '' + $(window).scrollTop() + header2_initialPadding);  
  }  
} else {  
  header2.css('padding-top', '' + header2_initialPadding);  
}
```

“Um algoritmo pode ser definido como uma sequência de passos que visam a atingir um objetivo bem definido”

```
render() {  
  return (  
    <React.Fragment>  
      <div className="py-5">  
        <div className="container">  
          <Title name="our" title</Title>  
          <div className="row">  
            <ProductConsumer>  
              {(value) => {  
                console.log</div>  
              }}  
            </ProductConsumer>  
          </div>  
        </div>  
      </div>  
    </div>  
  )  
}
```



# [Características]

- Especifica uma sequência de passos;
- Seus passos são ordenados de forma lógica;
- Apresentam ações claras e precisas;
- Fixam um padrão de comportamento;



# Exemplo{



1. Em um liquidificador, adicione a cenoura, os ovos e o óleo, depois misture.

}

2. Acrescente o açúcar e bata novamente por 5 minutos.

3. Em uma tigela ou na batedeira, adicione a farinha de trigo e depois misture novamente.

4. Acrescente o fermento e misture lentamente com uma colher.

}

5. Asse em um forno preaquecido a  $180^{\circ}$  C por aproximadamente 40 minutos.





# [Algoritmos Eficientes]

- Definir ações simples e sem ambiguidades;
- Organizar as ações de forma ordenada;
- Estabelecer uma sequência finita de passos.



# [Algoritmos Computacionais]

- Ler e escrever dados;
- Avaliar expressões algébricas, relacionais e lógicas;
- Tomar decisões com base em resultados de expressões;
- Repetir um bloco de ações de acordo com uma condição;
- ...muitas outras coisas...



# Algoritmos ◇ {

“Um algoritmo tem por objetivo representar mais fielmente o raciocínio envolvido na Lógica de Programação e, dessa forma, permite-nos abstrair de uma série de detalhes computacionais, que podem ser acrescentados mais tarde”

– **FORBELLONE, 2005.**

}



# [Construindo bons algoritmos]

- Passos sequenciais bem definidos;
- Quantidade passos finitos;
- Pode haver estruturas de controle ou de repetição;
- Pode haver estruturas de parada da execução;

&gt;

}



**uniesp**

Centro Universitário