



Computational Logic using Python - Unidade I

Olá, estudantes!

A **lógica computacional** é a base para a resolução de problemas, uma habilidade essencial para qualquer profissional de tecnologia. Ela permite que desenvolvam o raciocínio lógico necessário para resolver problemas complexos em partes menores, estruturadas e organizadas, facilitando sua solução. Além disso, a **programação em Python**, uma das linguagens mais populares e acessíveis, introduz conceitos essenciais de algoritmos e desenvolvimento de software. Python é amplamente utilizado em áreas como **ciência de dados, inteligência artificial, automação e desenvolvimento de sistemas e web**, permitindo aplicar os conceitos aprendidos em várias frentes da tecnologia.

Vem juntos mergulhar nesse universo!



Aula 1 - O que é Lógica Computacional?



Aula 2 - Conceitos de Programação em Python



Download do Conteúdo

Aula 1 - O que é Lógica Computacional?

Videoaula 1 - O que é Lógica Computacional?

Nesta aula, você vai descobrir o que é lógica computacional e por que ela é tão fundamental no campo da ciência da computação. A lógica computacional nos ajuda a estruturar o pensamento para resolver problemas de maneira eficiente e sistemática, sendo a base para a construção de algoritmos e o desenvolvimento de programas. Prepare-se para entender os princípios que orientam o funcionamento das máquinas e dos softwares que usamos todos os dias!



O que é Lógica Computacional?

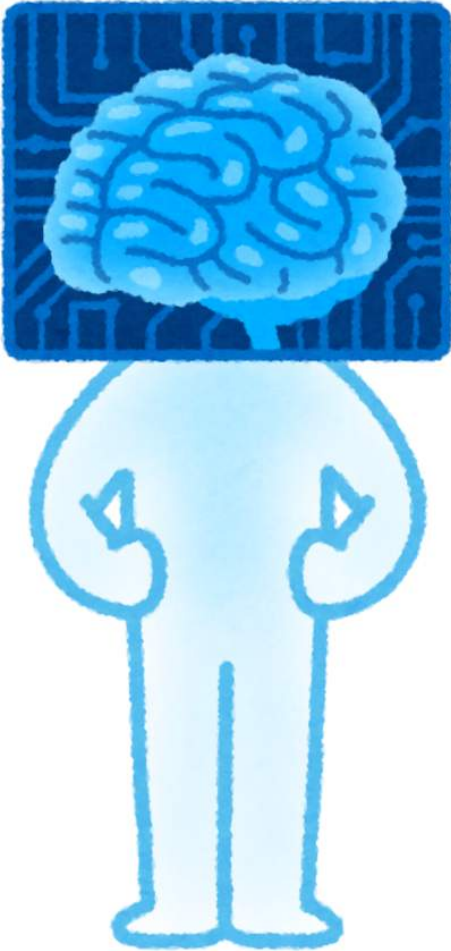


Ilustração fazendo a seguinte pergunta pergunta: O que é Lógica Computacional?

Resposta

Lógica computacional é um ramo da lógica e da ciência da computação que lida com a aplicação de princípios lógicos na

resolução de problemas computacionais. Ela envolve o uso de operações lógicas para criar soluções algorítmicas que podem ser processadas por computadores. Em essência, a lógica computacional estabelece as bases para desenvolver programas que funcionem de forma correta e eficiente.

Aplicações da Lógica Computacional

- **Desenvolvimento de Software:** A lógica é essencial para criar algoritmos que controlam o fluxo de um programa.
- **Inteligência Artificial:** Sistemas de IA dependem de modelos lógicos para tomar decisões e raciocinar.
- **Automação de Processos:** Automatizar tarefas envolve decisões condicionais e controle lógico para o fluxo de trabalho.

Algoritmo

Um **algoritmo** pode ser entendido como uma sequência finita de instruções bem definidas e ordenadas para realizar uma tarefa ou resolver um problema. Todo algoritmo precisa ser claro e não deve haver ambiguidade em suas etapas.

Características de um Algoritmo:

- **Precisão:** Deve ser composto de instruções claras e inequívocas;
- **Finitude:** Deve ter um número finito de etapas;
- **Entrada:** Recolhe dados necessários;
- **Saída:** Gera um resultado após o processamento;
- **Eficiência:** Deve ser executado no menor tempo e espaço possível.

Exemplos de Algoritmo

ALGORITMO PARA FAZER CAFÉ

1. Coloque água no bule;

2. Aqueça a água até ferver;
3. Adicione pó de café no filtro;
4. Despeje a água quente sobre o pó;
5. Aguarde a filtragem completa;
6. Sirva o café.

Entrada, Processamento e Saída

Esses três componentes formam a base do fluxo de execução de um algoritmo e de programas computacionais:

ENTRADA

PROCESSAMENTO

SAÍDA

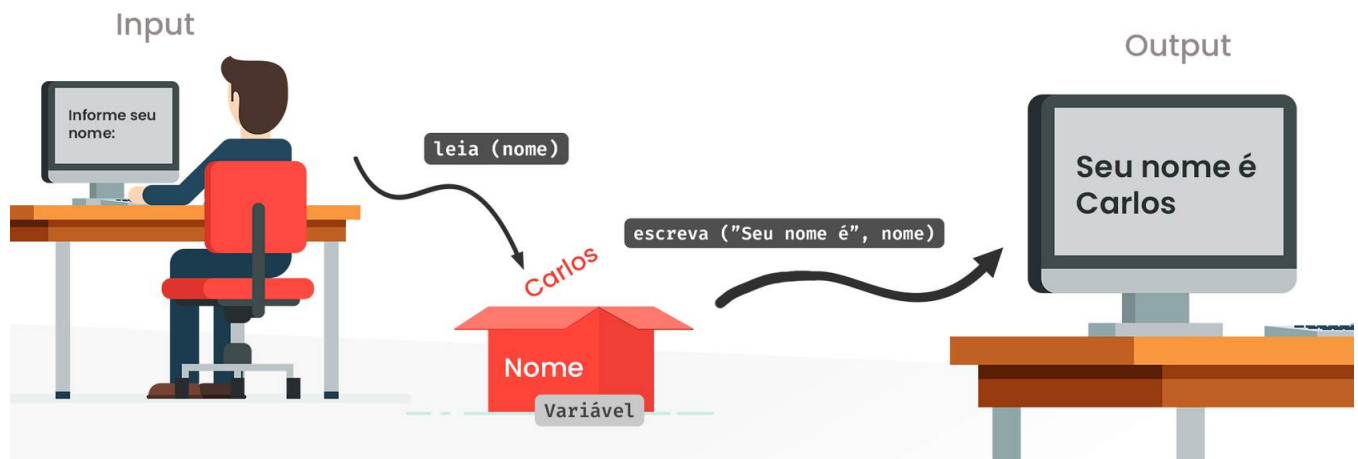
A **entrada** é qualquer dado que o programa recebe para realizar seu processamento. Ela pode ser fornecida pelo usuário, por outro sistema, ou por arquivos e bancos de dados.

| ENTRADA | PROCESSAMENTO | SAÍDA |
|--|---------------|-------|
| <p>O processamento é onde o algoritmo aplica operações lógicas e aritméticas nos dados de entrada. Esse processo inclui cálculos, comparações, tomadas de decisão e muito mais.</p> | | |

| ENTRADA | PROCESSAMENTO | SAÍDA |
|--|---------------|-------|
| <p>A saída é o resultado gerado pelo processamento. Ela pode ser exibida ao usuário, armazenada ou usada em outro processo.</p> | | |

Exemplo

Um programa que calcula a média de quatro notas terá como **entrada** as quatro notas fornecidas pelo usuário, o **processamento** será a soma das notas dividida por quatro, e a **saída** será a média calculada.



Descrição Narrativa

A descrição narrativa é uma forma de representar um algoritmo por meio de texto corrido, descrevendo cada etapa de maneira detalhada. Vamos usar o exemplo do cálculo da média de quatro notas como narrativa.

Exemplo: Cálculo da Média Aritmética de 4 Notas

1

Solicite ao usuário a primeira nota;

2

Solicite ao usuário a segunda nota;

- 3 Solicite ao usuário a terceira nota;
- 4 Solicite ao usuário a quarta nota;
- 5 Some as quatro notas;
- 6 Divida a soma por 4 para calcular a média;
- 7 Exiba o valor da média ao usuário.

Fluxograma (Diagrama de Blocos)

O **fluxograma** é uma representação gráfica do fluxo de um algoritmo, usando símbolos padronizados. Cada símbolo representa uma função no processo, como **decisão**, **processamento** ou **entrada/saída**.

Símbolos Comuns:

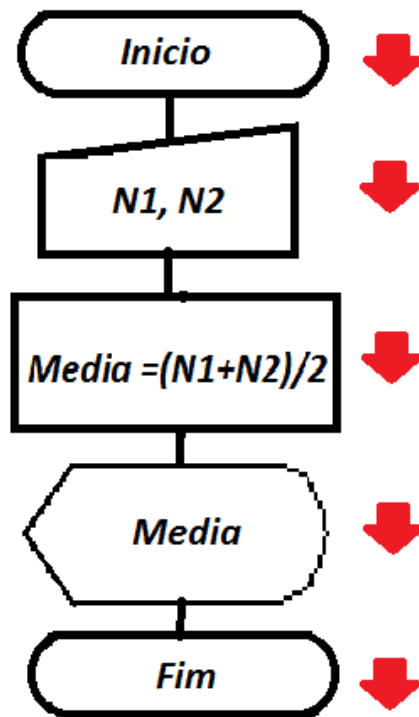
- **Elipse:** Início ou fim do processo;
- **Retângulo:** Etapas de processamento;

- **Paralelogramo:** Entrada ou saída de dados;
- **Losango:** Tomada de decisão.

EXEMPLO: FLUXOGRAMA PARA O CÁLCULO DA MÉDIA DE 2 NOTAS

O diagrama incluirá as seguintes etapas:

1. Início
2. Solicitar a primeira nota (entrada)
3. Solicitar a segunda nota (entrada)
4. Somar as 2 notas (processamento)
5. Dividir a soma por 2 (processamento)
6. Exibir a média (saída)
7. Fim



Videoaula 2 - Introdução ao Python

Nesta aula, você será introduzido ao universo da programação com Python, uma das linguagens mais simples e poderosas do mercado. Vamos explorar os primeiros passos para programar, desde a instalação do Python até a execução dos seus primeiros códigos. Conheça as principais características da linguagem, como sua sintaxe limpa e intuitiva, e descubra por que ela é uma das favoritas entre iniciantes e profissionais da área de tecnologia. Se você está começando sua jornada na programação, esta aula é o ponto de partida ideal!



Introdução ao Python

Python é uma das linguagens de programação mais populares para iniciantes, devido à sua sintaxe clara e simples. Ele é amplamente utilizado em várias áreas, incluindo ciência de dados, desenvolvimento web, automação e muito mais.

Principais Características do Python:

Sintaxe Simples: Código limpo e fácil de ler.

Interpretada: O código é executado linha por linha, o que facilita o processo de depuração.

Portabilidade: Pode ser executado em diferentes sistemas operacionais sem alterações.

EXEMPLO BÁSICO: "HELLO, WORLD!" EM PYTHON

```
print("Hello, World!")
```

EXEMPLO BÁSICO: "HELLO, WORLD!" EM PYTHON

```
print("Hello, World!")
```

CÓDIGO EM PYTHON: CÁLCULO DA MÉDIA ARITMÉTICA DE 4 NOTAS

Abaixo está o código para calcular a média aritmética de quatro notas em Python:

```
# Solicitar ao usuário as quatro notas
nota1 = float(input("Digite a primeira nota: "))
nota2 = float(input("Digite a segunda nota: "))
nota3 = float(input("Digite a terceira nota: "))
nota4 = float(input("Digite a quarta nota: "))

# Calcular a média aritmética
media = (nota1 + nota2 + nota3 + nota4) / 4

# Exibir o resultado
print("A média aritmética é:", media)
```

Explicação do Código

- 1 **Entrada:** O programa solicita as quatro notas ao usuário;
- 2 **Processamento:** A média é calculada somando as quatro notas e dividindo o resultado por quatro;
- 3 **Saída:** O valor da média é exibido ao usuário.

Siga para a próxima aula!

Aula 2 - Conceitos de Programação em Python

Videoaula 3 - Conceitos de Programação em Python

Se você está começando no mundo da programação, esta videoaula é essencial para você. Aqui, vamos abordar os conceitos fundamentais da programação usando a linguagem Python, uma das mais populares e versáteis. Entenda como variáveis, estruturas de controle e funções formam a base de qualquer programa, e comece a desenvolver suas próprias soluções de forma prática e eficiente.



Conceitos de Programação em Python

Variáveis e Constantes

Uma **variável** é um espaço de memória no qual podemos armazenar dados que podem ser modificados ao longo da execução do programa.

EXEMPLO

Se quisermos armazenar a idade de uma pessoa, podemos criar uma variável chamada `idade`.

```
idade = 25
```

Uma **constante** é um valor que, uma vez definido, não pode ser alterado durante a execução do programa. Em Python, não existe um tipo explícito de constante, mas por convenção, variáveis que não devem ser alteradas são escritas em letras maiúsculas.

EXEMPLO

```
PI = 3.1416
```

Dados x Informação



Ilustração fazendo a seguinte pergunta: O que são Dados?

Resposta

Dados são valores brutos, que por si só não têm significado, até serem processados. Eles podem ser números, palavras, medidas etc.

Exemplo de dado: "25" é um dado sem contexto.

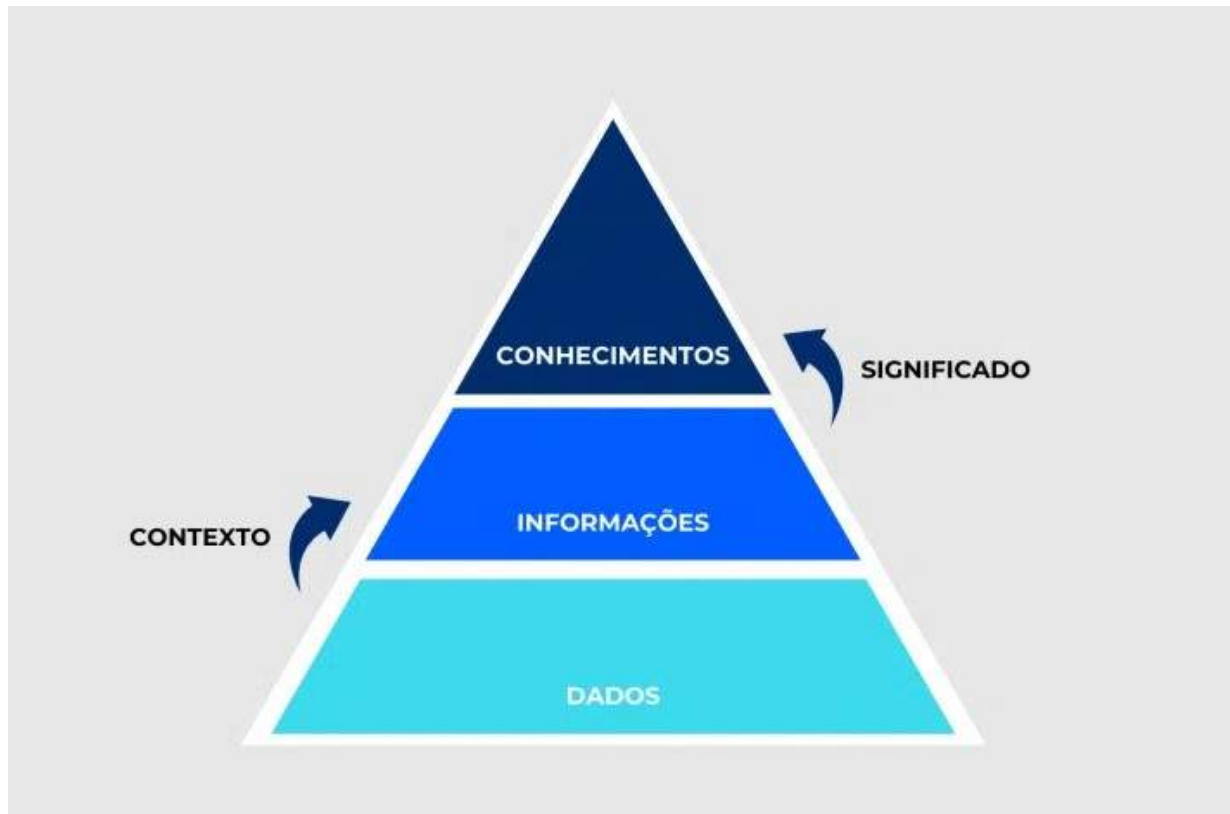


Ilustração fazendo a seguinte pergunta: O que é Informação?

Resposta

Informação é o resultado do processamento dos dados. Quando os dados são organizados e interpretados, eles se tornam informações.

Exemplo de informação: "25 anos" representa a idade de uma pessoa, transformando o dado em algo significativo.



Tipos de Dados no Python

Python suporta vários tipos de dados que ajudam a definir o tipo de valor que uma variável pode armazenar.

Alguns dos principais tipos de dados são:

NÚMEROS INTEIROS
(INT)

NÚMEROS DE PONTO
FLUTUANTE (FLOAT)

STRINGS (STR)

BOOLEANOS (BOOL)

Armazena valores inteiros, positivos ou negativos.

idade = 30

NÚMEROS INTEIROS
(INT)

NÚMEROS DE PONTO
FLUTUANTE (FLOAT)

STRINGS (STR)

BOOLEANOS (BOOL)

Armazena números decimais.

altura = 1.75

NÚMEROS INTEIROS
(INT)

NÚMEROS DE PONTO
FLUTUANTE (FLOAT)

STRINGS (STR)

BOOLEANOS (BOOL)

Armazena sequências de caracteres (texto).

```
nome = "Maria"
```

NÚMEROS INTEIROS
(INT)

NÚMEROS DE PONTO
FLUTUANTE (FLOAT)

STRINGS (STR)

BOOLEANOS (BOOL)

Armazena valores de Verdadeiro ou Falso.

```
is_student = True
```

Variáveis em Python

Em Python, uma variável é criada no momento em que atribuímos um valor a ela. Python é uma linguagem de tipagem dinâmica, o que significa que não precisamos declarar o tipo da variável explicitamente.

EXEMPLO DE CRIAÇÃO DE VARIÁVEL

```
nome = "Carlos"  
idade = 22  
altura = 1.82
```

Boas Práticas para Nomes de Variáveis

- Nomes de variáveis devem ser descritivos.
- Utilize **snake_case** para nomes compostos (ex: **data_nascimento**).

Videoaula 4 - Entrada e Saída no Python

Nesta aula, vamos aprofundar o estudo da interação entre o usuário e o programa, explorando os conceitos de entrada e saída em Python. Você aprenderá como coletar informações do usuário e como apresentar os resultados de forma clara e eficaz. Esses

conceitos são essenciais para criar programas dinâmicos e interativos, tornando sua experiência de programação ainda mais poderosa e interessante.



Entrada e Saída no Python

ENTRADA DE DADOS

SAÍDA DE DADOS

Para receber dados do usuário, utilizamos a função `input()`.

Exemplo de código:

```
nome = input("Digite seu nome: ")
```

Essa função pausa o programa e espera a entrada do usuário.

ENTRADA DE DADOS

SAÍDA DE DADOS

Para exibir dados na tela, usamos a função `print()`.

Exemplo de código:

```
print("Bem-vindo,", nome)
```

Operações Matemáticas em Python

Python permite a realização de diversas operações matemáticas básicas e avançadas com seus operadores embutidos.

Operadores Aritméticos

Os operadores aritméticos em Python incluem:

ADIÇÃO (+)

SUBTRAÇÃO (-)

MULTIPLICAÇÃO (*)

DIVISÃO (/)

EXPO

Soma dois valores.

```
soma = 5 + 3 # Resultado: 8
```

ADIÇÃO (+)

SUBTRAÇÃO (-)

MULTIPLICAÇÃO (*)

DIVISÃO (/)

EXPO

Subtrai um valor de outro.

```
subtracao = 10 - 4 # Resultado: 6
```

| | | | | |
|------------|---------------|-------------------|-------------|-------------------|
| ADIÇÃO (+) | SUBTRAÇÃO (-) | MULTIPLICAÇÃO (*) | DIVISÃO (/) | EXPONENCIAÇÃO (^) |
|------------|---------------|-------------------|-------------|-------------------|

Multiplica dois valores.

multiplicacao = 7 * 2 # Resultado: 14

| | | | | |
|------------|---------------|-------------------|-------------|-------------------|
| ADIÇÃO (+) | SUBTRAÇÃO (-) | MULTIPLICAÇÃO (*) | DIVISÃO (/) | EXPONENCIAÇÃO (^) |
|------------|---------------|-------------------|-------------|-------------------|

Divide um valor por outro.

divisao = 15 / 3 # Resultado: 5.0

| | | | | |
|------------|---------------|-------------------|-------------|-------------------|
| ADIÇÃO (+) | SUBTRAÇÃO (-) | MULTIPLICAÇÃO (*) | DIVISÃO (/) | EXPONENCIAÇÃO (^) |
|------------|---------------|-------------------|-------------|-------------------|

Eleva um valor à potência de outro.

```
potencia = 2 ** 3 # Resultado: 8
```

Exemplo Completo: Cálculo da Média

Agora que conhecemos as operações básicas, podemos aplicar no cálculo de uma média de quatro notas.

```
nota1 = float(input("Digite a primeira nota: "))
nota2 = float(input("Digite a segunda nota: "))
nota3 = float(input("Digite a terceira nota: "))
nota4 = float(input("Digite a quarta nota: "))

# Calcular a média
media = (nota1 + nota2 + nota3 + nota4) / 4

# Exibir o resultado
print("A média aritmética é:", media)
```

Parabéns, você finalizou esta unidade!

Download do Conteúdo

**Que tal baixar todo esse conteúdo e aproveitar os estudos
no seu melhor momento?**

Faça o download do conteúdo dessa unidade:



BAIXE O CONTEÚDO DESTA UNIDADE.pdf
2.3 MB

