



Raciocínio Algorítmico

2025

Prof. Dr. Fabrício Enembreck

Formação

- ◇ Graduado em Bacharelado em Informática – UEPG
- ◇ Mestrado – PUCPR
- ◇ Doutorado – UTC (Université de Technologie de Compiègne – França)

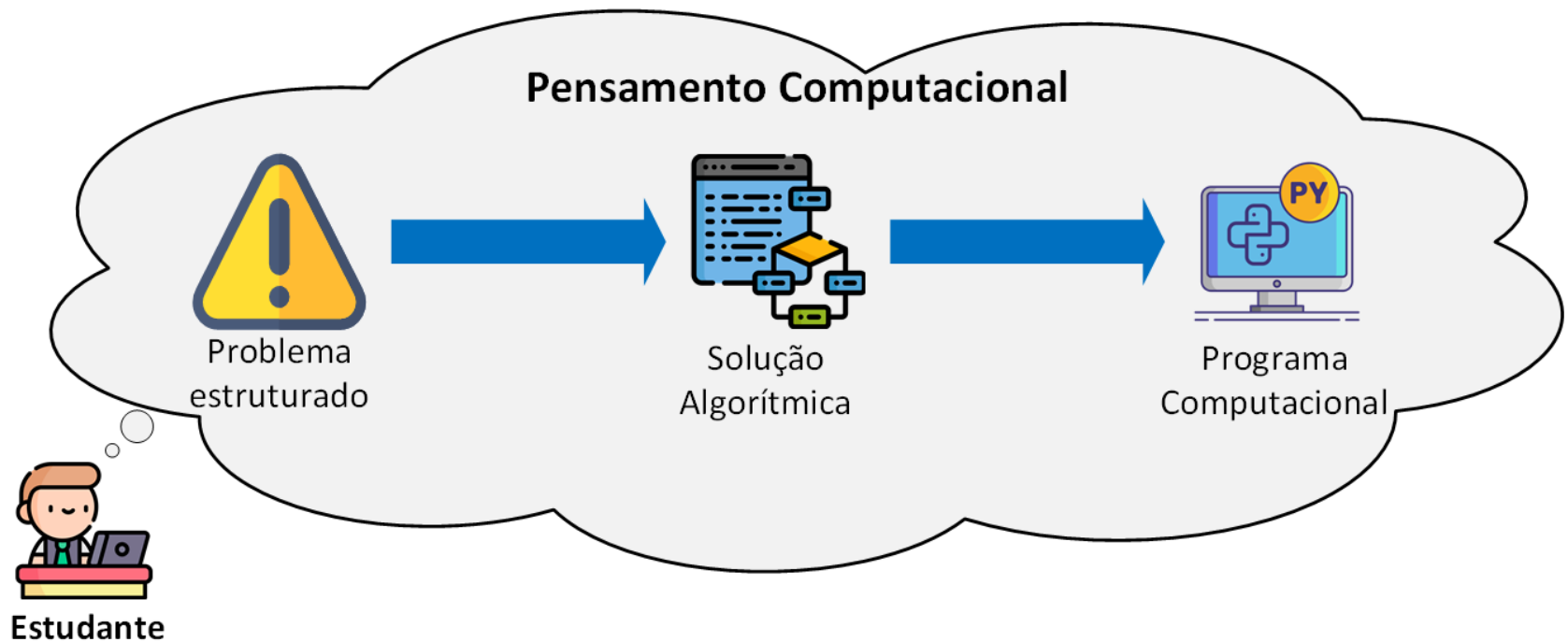
Atuação

- ◇ Inteligência Artificial
- ◇ Inteligência Artificial Distribuída
- ◇ Aprendizagem de Máquina
- ◇ Mineração de fluxos de dados

Pesquisa

- ◇ Programa de Pós-Graduação em Informática da PUCPR (PPGIa)
- ◇ Orientação de Mestrado e Doutorado
- ◇ Slides: prof.Vilmar Abreu

Mapa mental



Ementa

A disciplina de Raciocínio Algorítmico trata do **desenvolvimento do pensamento computacional** por meio da construção de **algoritmos**. Durante o semestre os estudantes aprendem a manipular variáveis, expressões lógico-aritmético-relacionais, estruturas de controle, estruturas de dados homogêneas e funções para a resolução de problemas computacionais na forma de algoritmos. Ao final da disciplina, o estudante é capaz de **implementar programas computacionais para problemas de baixa complexidade**, utilizando linguagem de programação, com autorregulação e atitude cooperativa.

Temas de Estudo

TE1: Algoritmos: conceitos, variáveis e tipos de dados; expressões lógico-aritmético-relacionais.

TE2: Estruturas de controle: seleção e repetição.

TE3: Estruturas de dados homogêneos: vetores e matrizes.

TE4: Funções e recursividade.

TE5: Boas práticas de programação.

Resultados de Aprendizagem 1/2

RA1. Projetar solução algorítmica, utilizando pensamento computacional, de forma autônoma.

ID1.1 Analisa o problema utilizando o pensamento computacional;

ID1.2 Elabora a solução algorítmica utilizando lógica de programação adequada ao problema;

ID1.3 Verifica se a solução algorítmica está aderente ao problema;

Resultados de Aprendizagem 2/2

RA2. Implementar programa computacional, respeitando boas práticas de programação

ID2.1 Codifica o programa computacional, utilizando os recursos adequados da linguagem de programação

ID2.2 Aplica boas práticas de programação na codificação do programa

ID2.3 Verifica se o programa computacional atende aos requisitos do problema

Metodologia e Avaliação

1. Diagnóstica
2. Formativa
3. Somativa
4. Recuperação

Datas importantes

Somativas	RA	Peso	Data Prevista
Prova Individual 1	RA1 e RA2	30%	23/04/2025
Projeto Colaborativo 1	RA1 e RA2	20%	24/04/2025
Recuperação 1	RA1 e RA2	-	07/05/2025
Prova Individual 2	RA1 e RA2	30%	11/06/2025
Projeto Colaborativo 2	RA1 e RA2	20%	12/06/2025
Recuperação 2	RA1 e RA2	-	18/06/2024

Plano de Recuperação (Covid)

Prezado(a) estudante(a), se você não pôde acompanhar as aulas presenciais em função de uma necessidade **de afastamento por COVID19 e entrou em contato com o SUEM e foi encaminhado para a realização de exercícios domiciliares**, você pode dar continuidade ao processo pedagógico de aprendizagem conforme o cronograma da disciplina. Siga as orientações de aprendizagem sugeridas pelo(a) seu/sua professor(a). **Não esqueça de realizar as entregas solicitadas conforme o plano de atividades para atribuição de presença.**

Programação

Identificar
um
problema



Planejar a
solução para
o problema



Implementar a
solução (algoritmo)
usando uma
linguagem de
programação



Algoritmo

- Sequência de passos, bem definido, que visa resolver um problema
- Desenvolver um algoritmo demanda:
 1. Saber como solucionar um problema
 2. Converter uma solução em um algoritmo
 3. Implementar este algoritmo em uma linguagem de programação
 4. Validar e testar o algoritmo

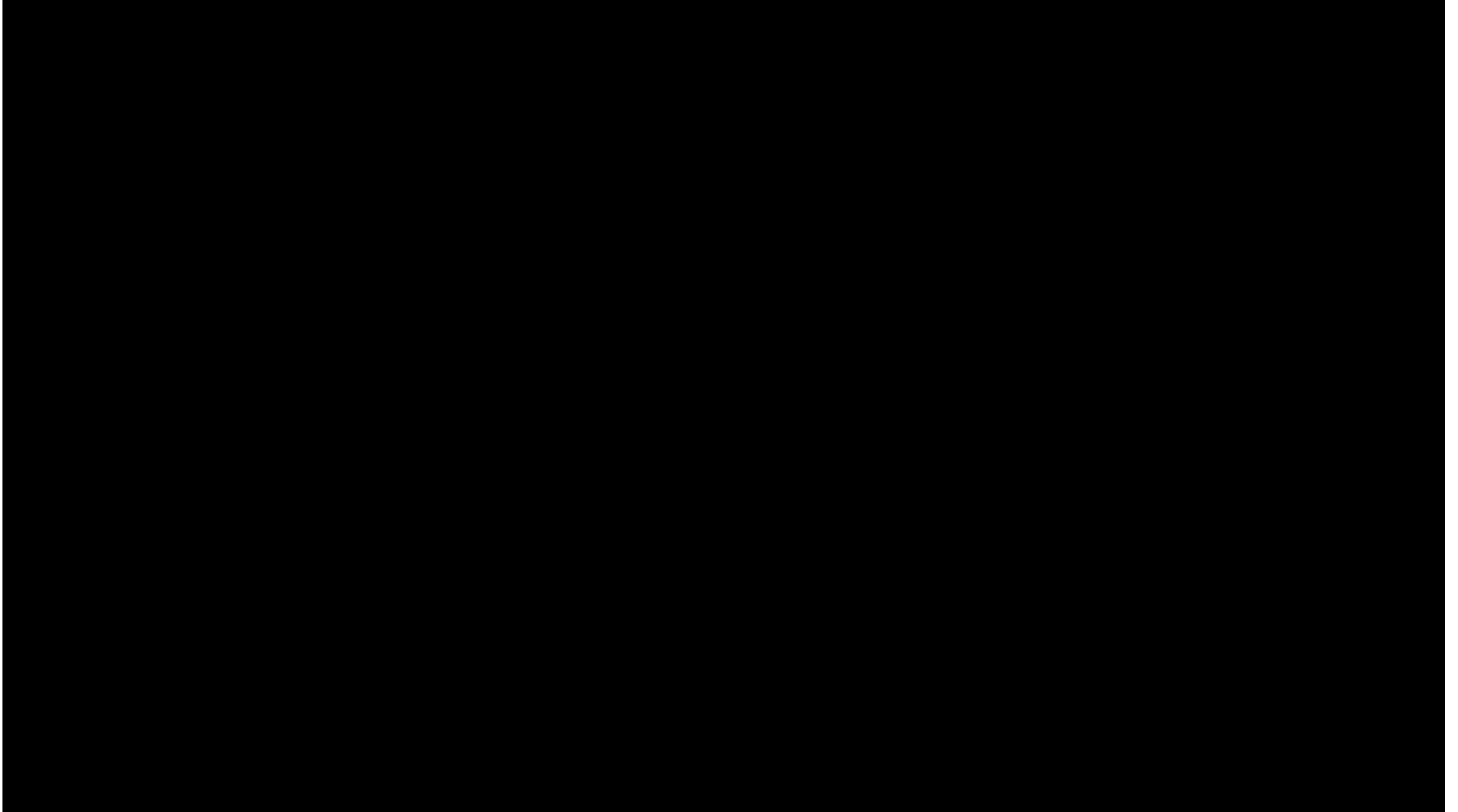
Exemplo: Como calcular o IMC?

- Descreva as etapas necessárias para calcular o IMC (Índice de Massa Corporal) de uma pessoa.

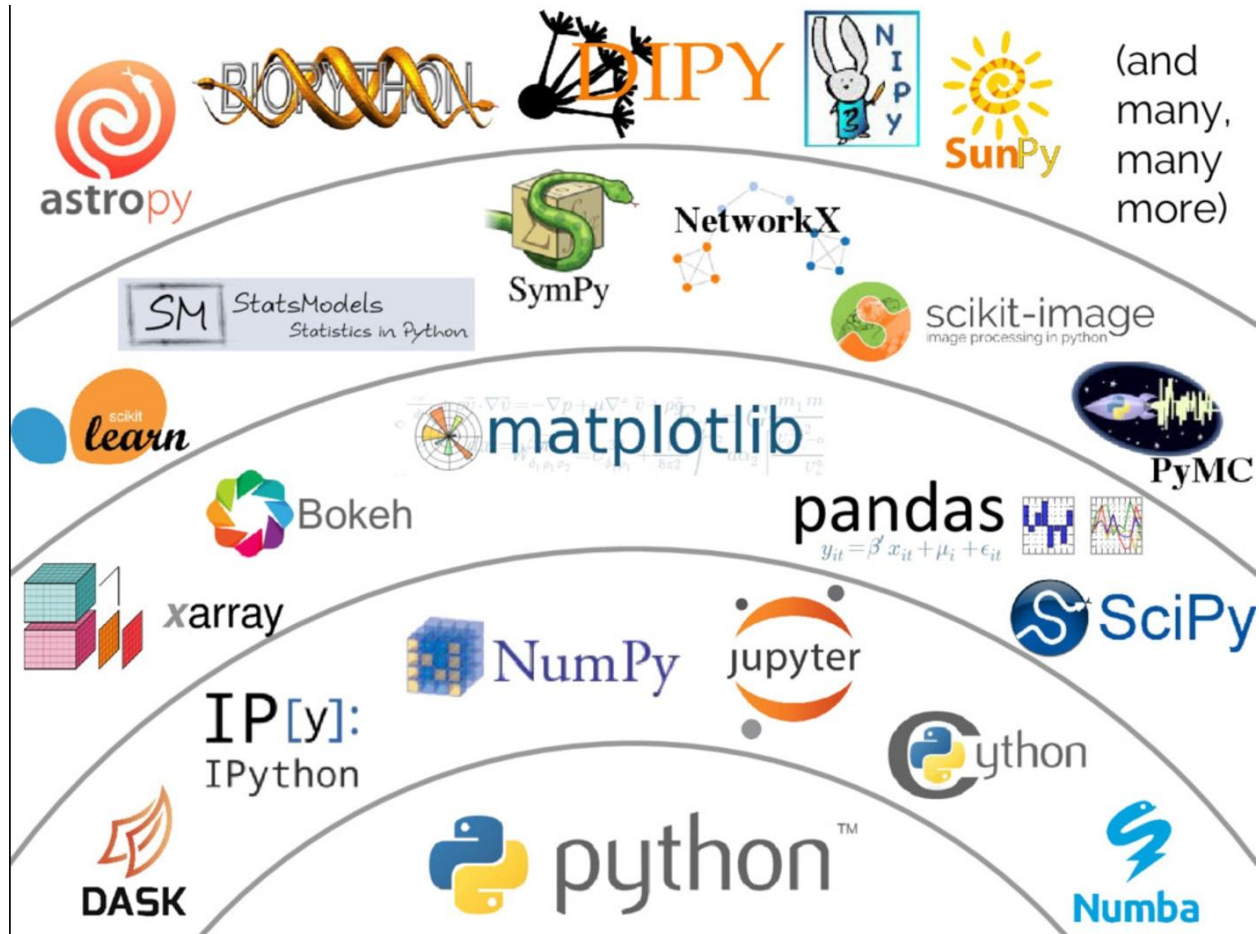
Linguagem de programação

- Infelizmente, se digitarmos qualquer solução proposta no computador, ele não vai conseguir executar o nosso algoritmo
- Por isso, precisamos “traduzir” nossas soluções para uma linguagem que o computador entenda
- Desta forma, uma linguagem de programação é uma linguagem utilizada para controlar máquinas
- Como linguagens humanas, nós temos regras sintáticas e semânticas para determinar estrutura e significado
- Note: linguagens de programação ainda são intermediárias, pois o computador entende apenas códigos binários (0s e 1s)

Linguagens de programação



Python



Ambiente



PyCharm 2019.3.2 (Community Edition)

Build #PC-193.6015.41, built on January 21, 2020

Runtime version: 11.0.5+10-b520.30 x86_64

VM: OpenJDK 64-Bit Server VM by JetBrains s.r.o

Powered by open-source software

THE DRIVE TO DEVELOP

Copyright © 2010–2020 JetBrains s.r.o.



Alternativa (temporária): <https://repl.it/languages/python3>

Variáveis

- Começam com letras ou _
- Não podem começar com número
- Apenas alfanuméricos (a-z, A-Z, 0-9 e _)
- Case-sensitive
- Evitar caracteres compostos (acentuados).

Variáveis (Boas práticas)

- Usar nomes significativos de variáveis
- Utilizaremos a prática conhecida como camelCase:

custoPassagem

valorHora

nomeCliente

pesoProduto

- Evitar:

x

variavel1

Tipos de dados

- **Inteiro (int)**: toda e qualquer informação numérica que pertença ao conjunto dos inteiros;
Ex: Idade de uma pessoa, número de alunos na sala
- **Real (float)**: toda e qualquer informação numérica que pertença ao conjunto dos reais;
Ex: Altura de uma pessoa, peso
- **Caracteres (string)**: toda e qualquer informação composta por um conjunto de caracteres alfanuméricos;
Ex: Nome, endereço
- **Lógico (bool)**: informação que assume somente dois valores possíveis (verdadeiro ou falso)
Ex: Porta aberta ou fechada, lâmpada acesa ou apagada

Entrada e Saída

Para que os algoritmos possam apresentar seus resultados e também possam obter dados (por exemplo solicitando-o ao usuário), comandos especiais de entrada e saída são necessários.

Entrada (dados do teclado)

- **variável = input(<texto informativo entre aspas>)**
- A variável receberá como conteúdo o valor lido, ou seja, digitado pelo teclado. O conteúdo lido é um dado dito “bruto”, do tipo string.
- **variável = int(input())**
Utilizado para ler um número inteiro do teclado.
- **variável = float(input())**
Utilizado para ler um número real do teclado.

Saída

- `print(variável)`
- `print("O nome é " + nome)`
- `print("A idade é " + str(idade))`

ATIVIDADE I

Crie um algoritmo que recebe quatro informações: e armazena em variáveis: Nome, CPF, Telefone e Ano de Nascimento.

Em seguida, imprima na tela esses valores formatados.



Expressões Aritméticas

- Operadores Aritméticos

Chamamos de operadores aritméticos o conjunto de símbolos que representa as operações básicas da matemática, a saber:

Operador	Função	Exemplos
----------	--------	----------

+	Adição	$2 + 3$, $x + y$
-	Subtração	$4 - 2$, $n - m$
*	Multiplicação	$3 * 4$, $a * b$
/	Divisão	$10 / 2$, $x1 / x2$

- Atribuição de variáveis

Para a atribuição de um valor à uma variável, utiliza-se o sinal de '=',

ATIVIDADE 2

Crie um algoritmo que calcula o IMC e imprime o resultado.



Exercícios

- 1) Escreva um algoritmo em Python para calcular a idade de alguém, sabendo-se seu ano de nascimento.
- 2) Escreva um algoritmo em Python para calcular o valor, em reais, que deve ser pago por um cliente de uma locadora de carros. Sabe-se que:
 - O valor de locação de cada carro é 100,00 reais;
 - O cliente pode locar um único carro por vários dias.
- 3) Leia do teclado a temperatura em Celsius e imprima o equivalente em Fahrenheit.

Fórmula: $(X \text{ }^{\circ}\text{C} \times 9/5) + 32$

Exercícios

- 4) Escrever um algoritmo para calcular a média de 4 notas.
- 5) Calcular sua idade em meses
- 6) Calcular preço de venda para produto por quilo

Bibliografia

FORBELLONE, André Luiz Villar; EBERSPÄCHER, Henri Frederico. Lógica de programação: a construção de algoritmos e estruturas de dados. 3. ed. São Paulo: Pearson Prentice Hall, 2005.

MANZANO, José Augusto N. G.; OLIVEIRA, Jayr Figueiredo de. Algoritmos: lógica para desenvolvimento de programação de computadores. 24. ed., rev. São Paulo: Érica, 2010.

MENEZES, Nilo Ney Coutinho. Introdução à programação com Python: algoritmos e lógica de programação para iniciantes. 1. ed. São Paulo: Novatec, 2010.



PUCPR
GRUPO MARISTA

Contato:

fabricio.enembreck@pucpr.br