



ALÉM DE JOGAR, EU FAÇO JOGOS!



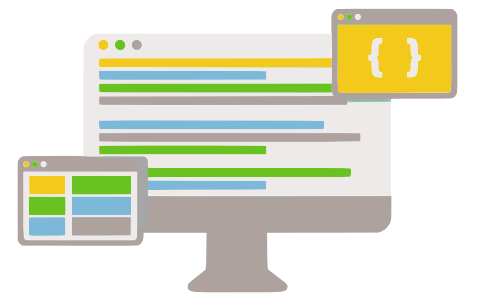
Material desenvolvido pela professora Juliana Oliveira para o projeto além de jogar eu faço jogos. Divulgação proibida.



História do Lua



Introdução a lógica de programação





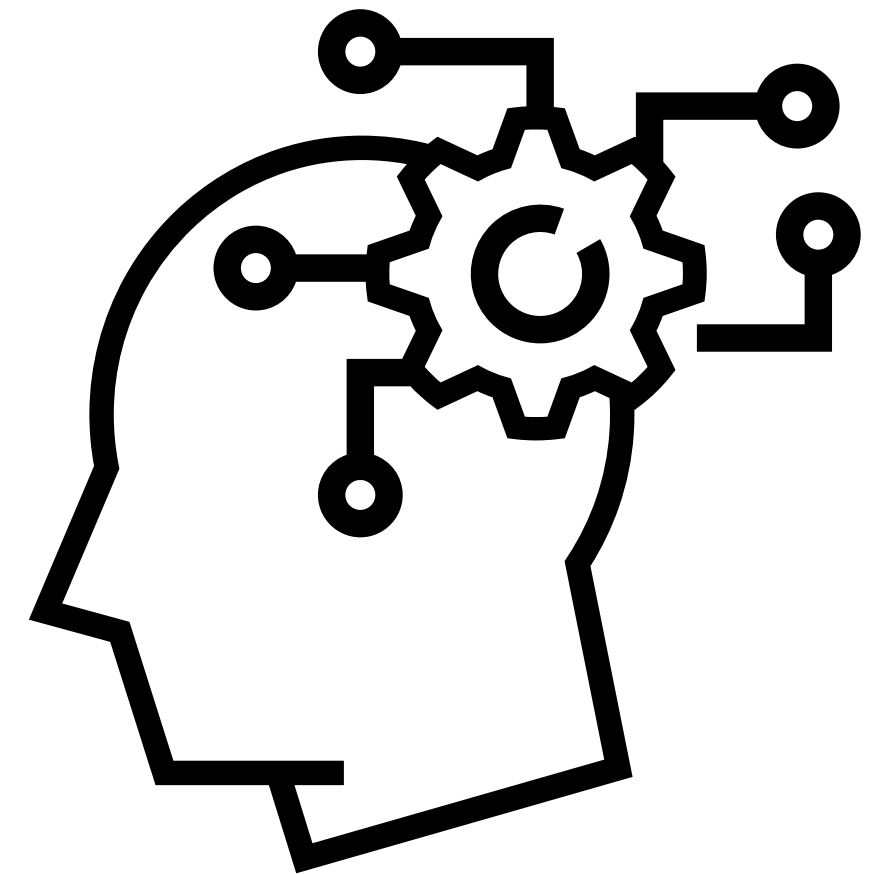
- ✓ Projetada por Tecgraf da Puc-Rio em 1993 para um projeto da Petrobrás.
- ✓ Linguagem orientada a objetos e dados e permite programação funcional.
- ✓ Primeiro jogo feito em Lua foi Grim Fandango em 1997.
- ✓ Alguns jogos famosos que utilizam Lua são Angry Birds, Counter-Strike Online, Roblox, Street Fighter IV.





Por quê precisamos falar sobre

A lógica de programação é necessária para quem quer trabalhar com desenvolvimento de sistemas e programas, é ela quem permite definir a sequência lógica para o desenvolvimento.



Lógica de programação é a técnica de encadear pensamentos para atingir determinado objetivo.



Sequência Lógica

Sequência lógica são passos executados até atingir um objetivo ou solução de um problema.

Instruções

Instruções são um conjunto de regras ou normas definidas para a realização ou emprego de algo. Em programação, é o que indica a um computador uma ação elementar a executar.



O que são?

Um Algoritmo é uma sequência finita de instruções bem claras, bem definidas e bem detalhadas que, quando colocadas em uma ordem lógica, nos levam a solução de um problema.

Os Algoritmos independem da linguagem de programação a ser utilizada. O objetivo da construção de Algoritmos é que eles possam ser adaptados a qualquer linguagem de programação.

Não existe apenas um Algoritmo para a solução de um problema. Um Algoritmo pode ser numérico ou não. Um Algoritmo pode ter tantas entradas e saídas de dados, quantas forem necessárias, pode até não ter entradas, mas obrigatoriamente deverá ter pelo menos uma saída de dados que é o resultado do processamento.



Ao montar um algoritmo, precisamos primeiro dividir o problema apresentado em três fases fundamentais.

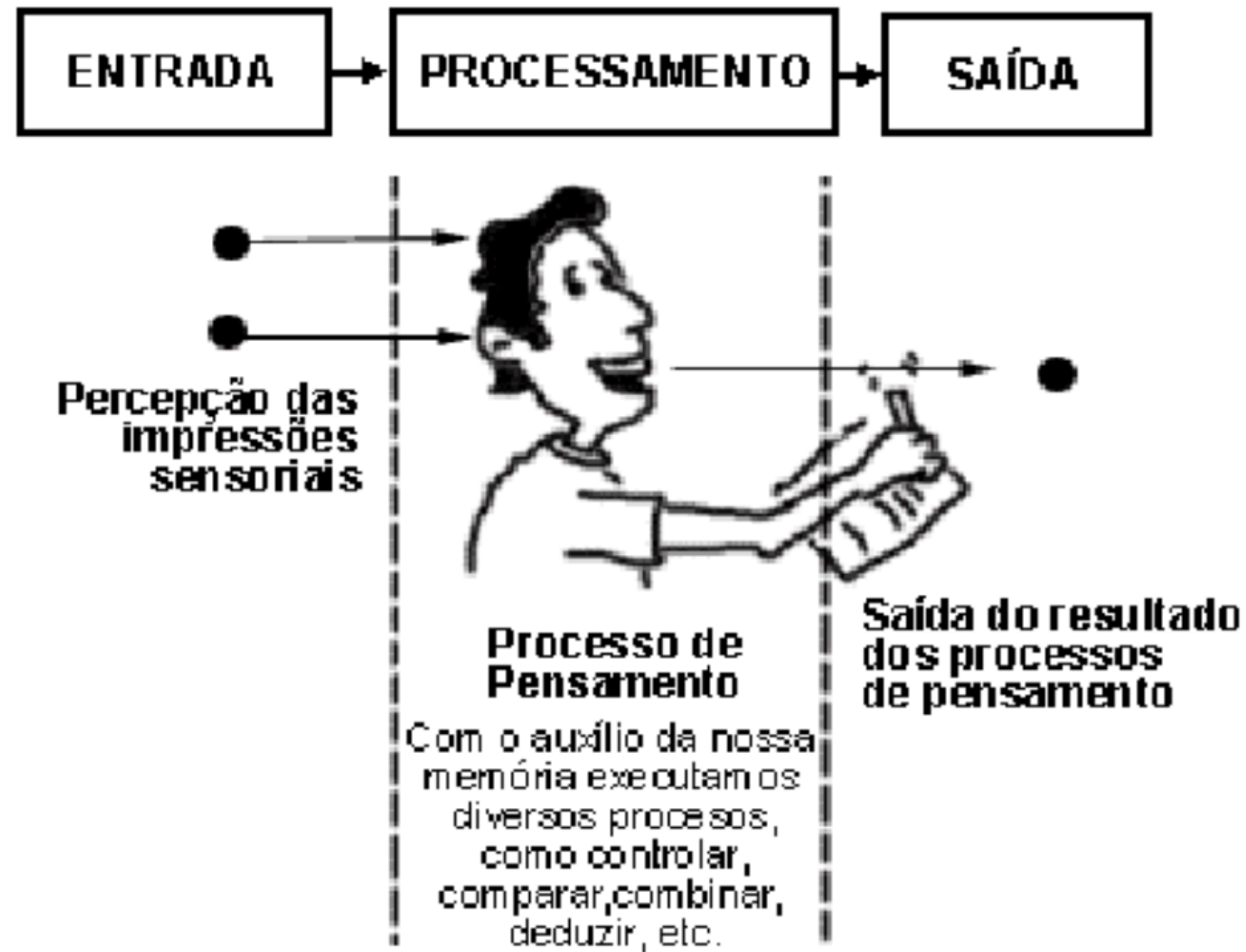


ENTRADA: São os dados de entrada do algoritmo

PROCESSAMENTO: São os procedimentos utilizados para chegar ao resultado final

SAÍDA: São os dados já processados

Exemplo de algoritmo





Para escrever um algoritmo precisamos descrever a sequência de instruções, de maneira simples e objetiva. Para isso utilizaremos algumas técnicas:

- Usar somente um verbo por frase
- Imaginar que você está desenvolvendo um algoritmo para pessoas que não trabalham com informática
- Usar frases curtas e simples
- Ser objetivo · Procurar usar palavras que não tenham sentido dúbio

Um algoritmo é uma das soluções e não “a solução” de um problema.



- Finitude: um algoritmo deve sempre terminar após um número finito de passos, ou seja, deve ter fim.
- Definição: cada passo de um algoritmo deve ser precisamente definido, sem dar margem à dupla interpretação (não ambíguo).
- Entradas: capacidade de receber dados do mundo exterior.
- Saídas: poder gerar informações sobre o resultado para o mundo exterior.
- Eficácia: os passos de um algoritmo devem conduzir à solução do problema proposto.



Exemplo 1 - Chupar uma bala:

- Pegar a bala
- Retirar o papel
- Chupar a bala
- Jogar o papel no lixo

Exemplo 2 - Batata-frita:

Início

- Pegar as batatas
- Descascar as batatas
- Cortar as batatas em tiras
- Pegar uma panela

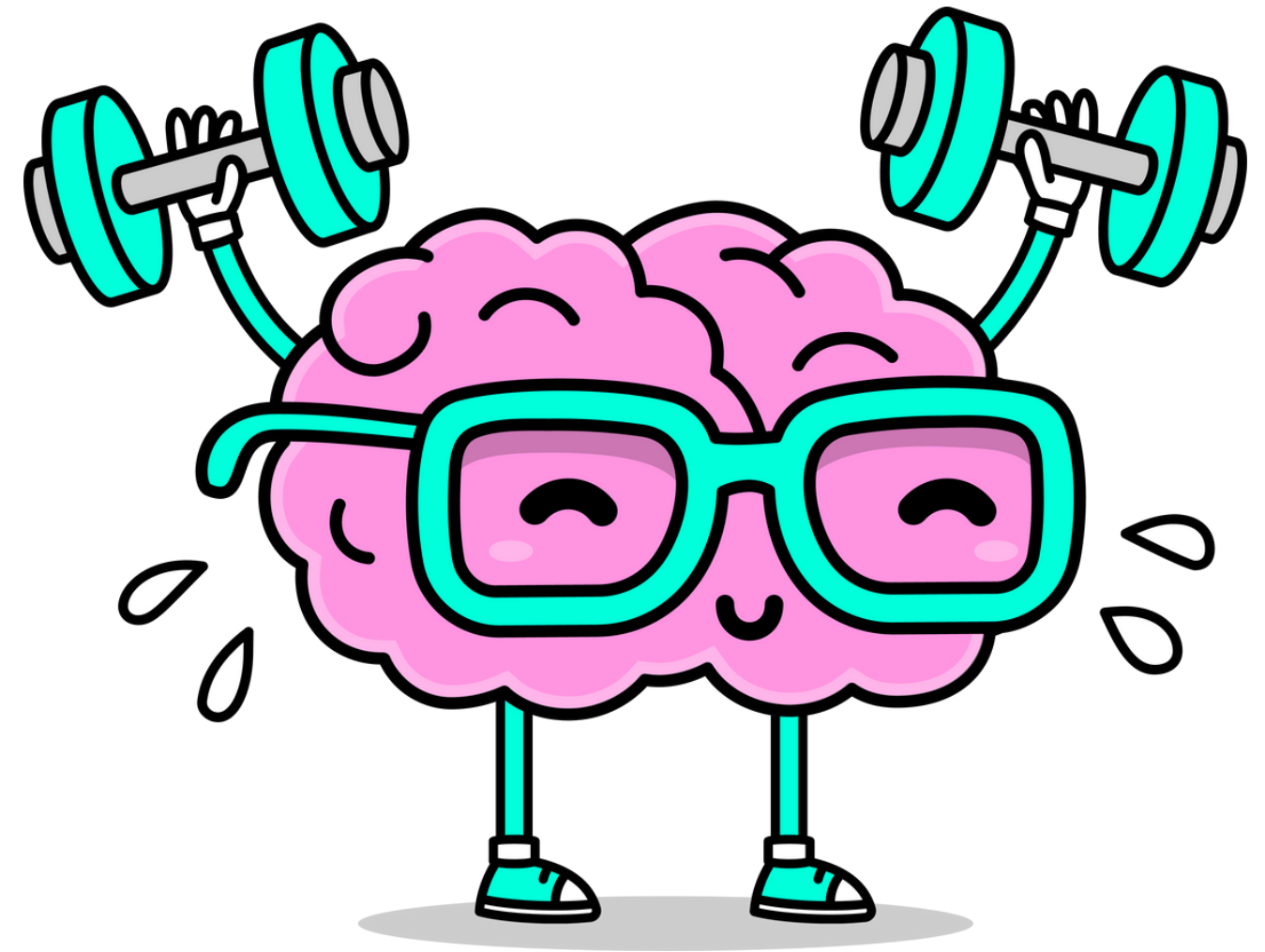
- Colocar óleo na panela
- Acender o fogo do fogão
- Colocar a panela com óleo para esquentar
- Esperar o óleo esquentar
- Colocar as batatas para fritar
- Aguardar o tempo ideal até que as batatas estejam fritas
- Remover as batatas-fritas e servir

Fim



Exercícios

1. Crie um algoritmo de alguma tarefa que você executa no dia a dia.
2. Faça um algoritmo para trocar uma lâmpada.
Descreva com detalhes





Constantes:

1. Numérico
2. Data
3. Lógica
4. Caracter (texto, literal, string).

Variáveis:

1. Numérico
2. Data
3. Lógica
4. Caracter (texto, literal, string).

Expressões:

1. Operandos
2. Operadores



Variáveis e constantes são os elementos básicos que um programa manipula.

Constantes:

São valores utilizados no processamento, mas que não sofrerão alterações durante o mesmo. Exemplos: nomes, datas de nascimento, números de funcionários, nomes de cidades, etc.

Variáveis:

Variáveis são endereços de memória (Ram) que armazenam temporariamente valores. É a representação simbólica dos elementos de um certo conjunto. Cada variável corresponde a uma posição de memória, cujo conteúdo pode ser alterado ao longo do tempo durante a execução de um programa. As variáveis são declaradas no início do algoritmo.



Em Lua, usamos constantes caracter (texto e literal) ou alfanuméricas (mistura de letras e números).

Os valores armazenados nas variáveis permanecem até que seja atribuído um novo valor para a mesma variável.

As variáveis podem ser globais ou locais, onde as globais são vistas por todos os módulos dos programas e as locais somente nas rotinas onde são criadas.



Os operadores são meios pelo qual incrementamos, decrementamos, comparamos e avaliamos dados dentro do computador. Temos três tipos de operadores:

- Operadores Aritméticos
- Operadores Relacionais
- Operadores Lógicos

Operadores aritméticos

Os operadores aritméticos são os utilizados para obter resultados numéricos. Além da adição, subtração, multiplicação e divisão, podem utilizar também o operador para exponenciação.



Operadores relacionais

Os operadores relacionais são utilizados para comparar String de caracteres e números. Os valores a serem comparados podem ser caracteres ou variáveis. Estes operadores sempre retornam valores lógicos (verdadeiro ou falso/ True ou False).

Para estabelecer prioridades no que diz respeito a qual operação executar primeiro, utilize os parênteses. Os operadores relacionais são:

Descrição	Símbolo
Igual a	=
Diferente de	<> ou #
Maior que	>
Menor que	<
Maior ou igual a	>=
Menor ou igual a	<=



Exemplo:

Tendo duas variáveis $A = 5$ e $B = 3$

Os resultados das expressões seriam:

Expressão	Resultado
$A = B$	Falso
$A \neq B$	Verdadeiro
$A > B$	Verdadeiro
$A < B$	Falso
$A \geq B$	Verdadeiro
$A \leq B$	Falso



Operadores lógicos

Os operadores lógicos servem para combinar resultados de expressões, retornando se o resultado final é verdadeiro ou falso.

Os operadores lógicos são:

E	AND
OU	OR
NÃO	NOT

E / AND: Uma expressão AND (E) é verdadeira se todas as condições forem verdadeiras.

OR/OU: Uma expressão OR (OU) é verdadeira se pelo menos uma condição for verdadeira.

NOT: Um expressão NOT (NÃO) inverte o valor da expressão ou condição, se verdadeira inverte para falsa e vice-versa.



A tabela abaixo mostra todos os valores possíveis criados pelos três operadores lógicos (AND, OR e NOT)

1º Valor	Operador	2º Valor	Resultado
T	AND	T	T
T	AND	F	F
F	AND	T	F
F	AND	F	F
T	OR	T	T
T	OR	F	T
F	OR	T	T
F	OR	F	F
T	NOT		F
F	NOT		T



Exemplo:

Suponha que temos três variáveis $A = 5$, $B = 8$ e $C = 1$
Os resultados das expressões seriam:

Expressões			Resultado
$A = B$	AND	$B > C$	Falso
$A <> B$	OR	$B < C$	Verdadeiro
$A > B$	NOT		Verdadeiro
$A < B$	AND	$B > C$	Verdadeiro
$A >= B$	OR	$B = C$	Falso
$A <= B$	NOT		Falso



Exercícios

1. Sabendo que $A=3$, $B=7$ e $C=4$, informe se as expressões ao lado são verdadeiras ou falsas.

- a) $(A+C) > B$
- b) $B \geq (A + 2)$
- c) $C = (B - A)$
- d) $(B + A) \leq C$
- e) $(C+A) > B$

2. Sabendo que $A=5$, $B=4$ e $C=3$ e $D=6$, informe se as expressões ao lado são verdadeiras ou falsas.

- a) $(A > C)$ **AND** $(C \leq D)$
- b) $(A+B) > 10$ **OR** $(A+B) = (C+D)$
- c) $(A \geq C)$ **AND** $(D \geq C)$