

# Aula 01 - Lógica

## Algoritmos

Uma boa definição do que é um algoritmo é de que o mesmo trata-se de “uma sequência finita de passos para solucionar um problema”. Executamos algoritmos em nosso dia-a-dia a todo tempo, como por exemplo, sair de casa e ir até o trabalho, realizar compras em um mercado, os passos desde o momento em que acordamos até chegarmos à parada de ônibus, etc.

Mesmo para essas questões do cotidiano, podem-se ter diversos algoritmos para solucionar o mesmo problema, e que muitas vezes, estes só funcionariam dependendo de fatores externos. Como fazer um bolo de chocolate à noite visto que faltou luz? A solução provavelmente será diferente da solução convencional.

## Método para a construção de algoritmos

Para a construção de qualquer tipo de algoritmo, é necessário seguir estes passos:

1. Entender completamente o problema a ser resolvido, destacando os pontos mais importantes.
2. Definir os dados de entrada
3. Criar um passo-a-passo para a solução do problema.

O computador pode auxiliá-lo em qualquer tarefa. Ele tem um processamento muito rápido e não cansa, porém ele não tem iniciativa e de certa forma não é inteligente, precisa ser programado para executar suas tarefas e para isso utilizamos a lógica de programação.

## Lógica

Normalmente utiliza-se a palavra “lógica” no dia-a-dia para indicar algo coerente e racional. Quando se encadeia alguns pensamentos e chega-se a uma conclusão, pode-se ao final concluir.

Pode-se dizer que a lógica seria a “correção do pensamento”, pois uma de suas preocupações é determinar quais operações são válidas e quais são inválidas.

### Exemplos

a) Todo mamífero é um animal.

Todo cachorro é um mamífero.

Portanto, todo cachorro é um animal

b) José é mais alto que Maria.

João é mais alto que José.

Portanto, João é mais alto que Maria.

c) Todos os filhos de João são mais altos do que Maria  
Antônio é filho de João

### ***O que pode-se concluir logicamente?***

Como Antônio é filho de João, e os filhos de João são mais altos que Maria, logo, Antônio é mais alto que Maria. Com a mesma sentença e usando lógica, verifica-se também que Maria é menor do que Antônio, além disso, nada pode-se concluir entre a altura de Maria e João.

Esses exemplos representam um argumento composto de duas premissas e uma conclusão e estabelece uma relação que pode ser válida ou não. Esse é um dos objetivos da lógica. Outro desses objetivos é criar uma representação mais formal de apresentar premissas e conclusões, de forma diferente da linguagem natural que muitas vezes depende de interpretação.

Essa mesma sequência lógica de pensamentos, descritos em uma linguagem formal, chegando a conclusões válidas e inválidas será utilizada na lógica de programação para se criar programas de computador.

## **Representação de um algoritmo**

Algoritmos na verdade são linhas de raciocínio e podem ser representados de diversas maneiras, tanto de forma textual como em uma lista de instruções/ações utilizando o português, quanto de forma gráfica em formato de diagramas.

Nos diagramas, as palavras são substituídas por formas geométricas representando instruções e opções dentro do algoritmo. Segue exemplo de um algoritmo que descreve a troca de uma lâmpada.

### **Algoritmo - Troca de uma lâmpada**

- A lâmpada não ligou
- se a lâmpada não estiver plugada
  - plugar a lâmpada;
- senão
  - se o bulbo queimou
  - pegar uma escada
  - colocar a escada embaixo da lâmpada
    - trocar o bulbo
  - senão
    - comprar uma lâmpada nova

Exemplos:

### **Troca de resistência com teste e repetições**

- ligar o chuveiro

- **se o chuveiro não esquentar, então**
- pegar uma escada;
- posicionar a escada embaixo do chuveiro
- buscar uma resistência nova
- subir na escada
- retirar a resistência queimada
- colocar a resistência nova
- **se o chuveiro não esquentar, então**
  - retirar a resistência queimada
  - colocar a resistência nova
  - **se o chuveiro não esquentar, então**
    - retirar a resistência queimada
    - colocar a resistência nova
    - **se o chuveiro não esquentar, então**
      - retirar a resistência queimada
      - colocar a resistência nova
      - .
      - .
      - .
      - **Até quando?**

É importante notar que o algoritmo acima não está terminado, falta determinar até quando será feito o teste do chuveiro. A execução do algoritmo deve terminar quando se conseguir colocar uma resistência que faça o chuveiro esquentar, caso contrário, os testes serão executados indefinidamente. Essa solução está mais próxima do objetivo, pois garante que o chuveiro volte a esquentar, mesmo que sejam trocadas várias resistências. Porém não se sabe qual vai ser o número exato de trocas.

Inspecionando o algoritmo, é fácil identificar que o teste da nova resistência é realizado por esse conjunto de instruções:

- **se o chuveiro não esquentar, então**
  - **retirar a resistência queimada**
  - **colocar a resistência nova**

Então, ao invés de se escrever várias vezes essas instruções, pode-se alterar o algoritmo para que a cada ação de colocar uma resistência nova, seja realizado o teste do chuveiro, fazendo com que essas instruções sejam executadas quantas vezes forem necessárias sem a necessidade de reescrevê-las.

Agora será expressa essa repetição de instruções sem repetir o texto da instrução e definir um limite para essa repetição para garantir uma condição de parada do algoritmo. Segue uma possível solução:

- **enquanto** o chuveiro não esquentar, **faça**
  - retirar a resistência queimada
  - colocar uma resistência nova

A condição “chuveiro não esquentar” permaneceu e criou-se um fluxo repetitivo que será finalizado assim que a condição de parada for falsa, ou seja, assim que o chuveiro esquentar. Dessa forma o número de repetições ficou indefinido, porém é finito, e depende apenas da condição de parada que for criada. Assim a troca de resistência será feita até que o objetivo seja atingido, ou seja, o chuveiro esquentar.

### **Troca de resistência com teste e condição de parada**

- ligar o chuveiro
- **se** o chuveiro não esquentar, **então**
  - pegar uma escada;
  - posicionar a escada embaixo do chuveiro
  - buscar uma resistência nova
  - subir na escada
  - retirar a resistência queimada
  - colocar a resistência nova
  - **enquanto** o chuveiro não esquentar, **então**
    - retirar a resistência queimada
    - colocar a resistência nova

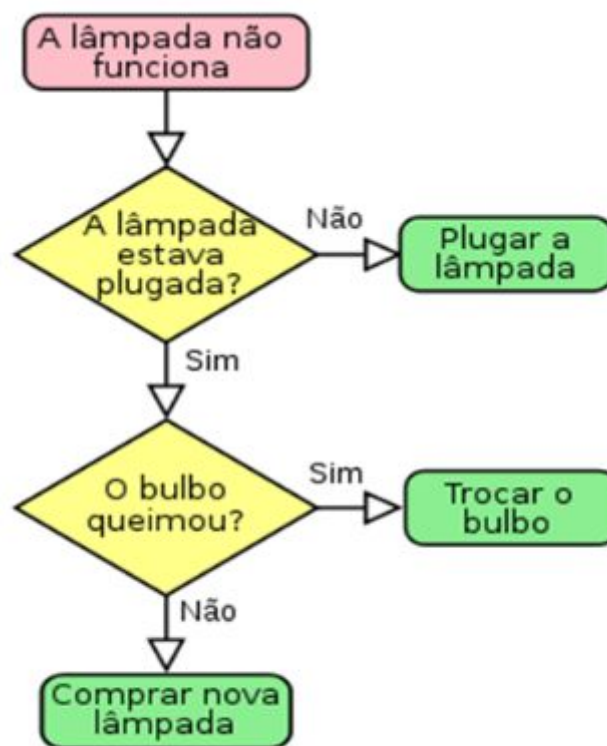
No primeiro algoritmo para resolver o problema da troca da resistência, havia um pequeno conjunto de instruções que deveriam ser executadas, passo a passo, uma após a outra, fazendo uma ordem sequencial de execução, a estrutura sequencial.

Porém, nem sempre todas as instruções previstas deveriam ser executadas, ou seja, algumas instruções não deveriam ser executadas dependendo de uma condição, assim construiu-se uma primeira estrutura de seleção, através de um teste condicional que permita ou não que o fluxo de execução aconteça.

**Caso a troca da resistência não tenha ocorrido com sucesso, o algoritmo desenvolvido vai repetir as instruções que efetuam uma nova troca de resistência, até que isso ocorra com sucesso. Dessa forma essa sequência de passos resolve o problema da troca da resistência, portanto pode-se considerar que é, de fato, um algoritmo para resolução do problema da troca da resistência.**

## Fluxogramas

As formas gráficas são menos dependentes de interpretação, deixando mais claro o que deve ocorrer em cada um dos passos (instruções) e principalmente o fluxo de execução. Porém, como em uma linguagem de programação, é necessário conhecer o significado de cada uma das formas geométricas utilizadas no diagrama e qual a ação esperada após sua execução. A figura abaixo apresenta um diagrama de fluxo do algoritmo de **troca de uma lâmpada**.

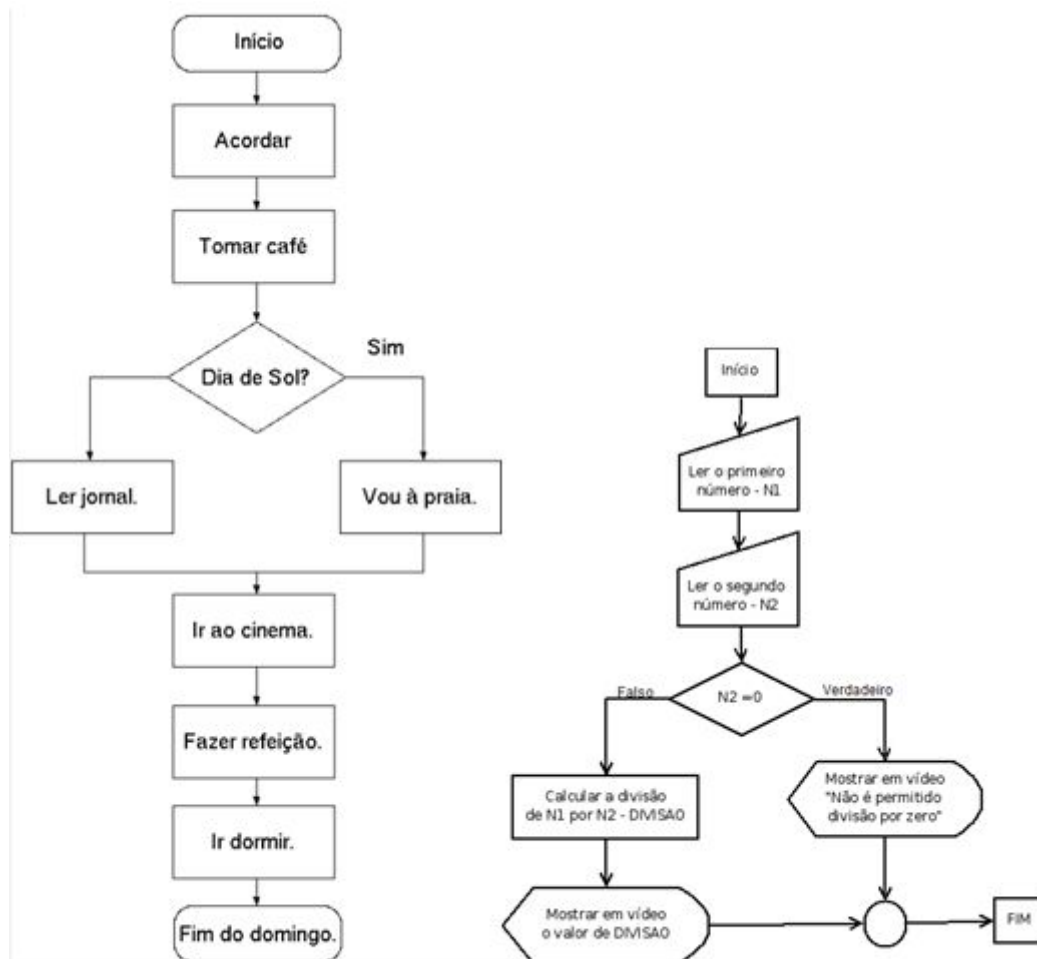


**Fluxograma** é uma representação de um processo que utiliza símbolos gráficos para descrever passo a passo a natureza e o fluxo deste processo. Nesse diagrama existem formas geométricas que indicam ação e formas que indicam uma decisão a ser tomada com base em uma informação lógica.



Tanto a representação textual utilizando português, como a representação gráfica de um algoritmo utilizando fluxograma tem vantagens e desvantagens. Algoritmos em português são mais fáceis de criar, porém seguidamente possuem ambiguidades próprias da nossa língua. Algoritmos descritos com auxílio de fluxogramas deixam bem mais claros o fluxo de execução das instruções, porém é necessário “desenhá-lo” ou lançar mão de aplicações para auxiliar nesse desenho.

Exemplos:



Responda as questões e envie por email de acordo com as orientações definidas no post do Classroom

1. O que é um algoritmo?
2. O que é um algoritmo de programação?
3. Como pode ser representado um algoritmo de programação?
4. Resolva os desafios lógicos abaixo:

Três senhoras – dona Branca, dona Rosa e dona Violeta – passeavam pelo parque quando dona Rosa disse:

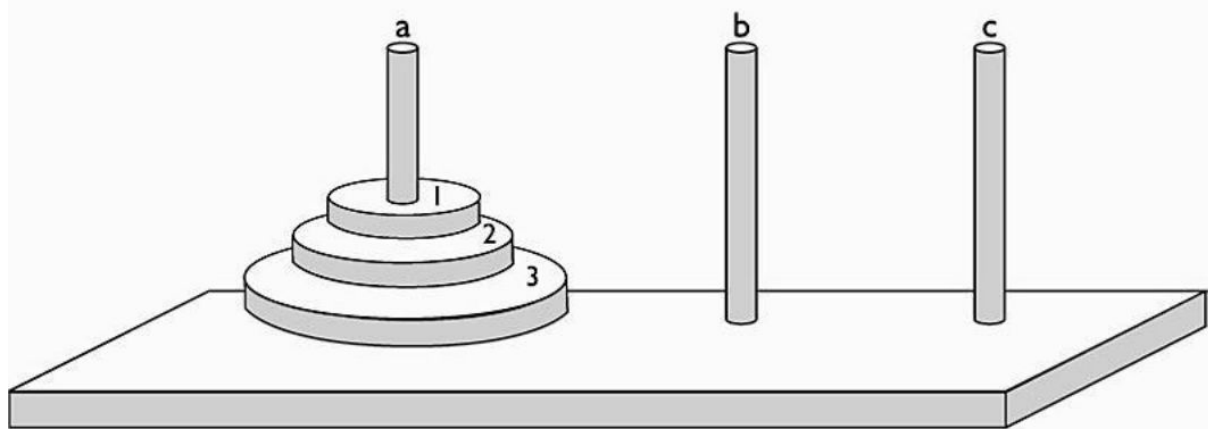
– Não é curioso que estejamos usando vestidos de cores branca, rosa e violeta, embora nenhuma de nós esteja usando um vestido de cor igual ao seu próprio nome?

– Uma simples coincidência – respondeu a senhora com o vestido violeta.

Qual a cor do vestido de cada senhora ?

Um homem precisa atravessar um rio com um barco que possui capacidade apenas para carregar ele mesmo e mais uma de suas três cargas, que são: um lobo, um bode e um maço de alfafa. O que o homem deve fazer para conseguir atravessar o rio sem perder suas cargas? Escreva um algoritmo mostrando a resposta, ou seja, indicando todas as ações necessárias para efetuar uma travessia segura.

Elabore um algoritmo que mova três discos de uma Torre de Hanói, que consiste em três hastes ( $a - b - c$ ), uma das quais serve de suporte para três discos de tamanhos diferentes ( $1 - 2 - 3$ ), os menores sobre os maiores. Pode-se mover um disco de cada vez para qualquer haste, contanto que nunca seja colocado um disco maior sobre um menor. O objetivo é transferir os três discos para outra haste.



Considere que uma calculadora comum, de quatro operações, está com as teclas de divisão e multiplicação inoperantes. Escreva algoritmos que resolvam as expressões matemáticas a seguir usando apenas as operações de adição e subtração.

- a)  $12 \times 4$
- b)  $23 \times 11$
- c)  $10 \div 2$
- d)  $175 \div 7$
- e)  $2^8$