

Algoritmo

Um algoritmo é formalmente uma sequência finita de passos que levam a execução de uma tarefa. Podemos pensar em algoritmo como uma receita, uma sequência de instruções que dão cabo de uma meta específica. Estas tarefas não podem ser redundantes nem subjetivas na sua definição, devem ser claras e precisas.

Como exemplos de algoritmos podemos citar os algoritmos das operações básicas (adição, multiplicação, divisão e subtração) de números reais decimais. Outros exemplos seriam os manuais de aparelhos eletrônicos, como um videocassete, que explicam passo-a-passo como, por exemplo, gravar um evento. Até mesmo as coisas mais simples, podem ser descritas por sequências lógicas. Por exemplo:

“Chupar uma bala”.

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

Outro Exemplo de algoritmo

Imagine o trabalho de um recepcionista de cinema, ele deve conferir os bilhetes e direcionar o cliente para a sala correta. Além disso, se o cliente estiver 30 minutos adiantado o recepcionista deve informar que a sala do filme ainda não está aberta. E quando o cliente estiver 30 minutos atrasado o recepcionista deve informar que a entrada não é mais permitida (Nota: Essas regras não são 100% verdade, eu as defini neste artigo apenas para fins didáticos). Vamos escrever um algoritmo para descrever a atividade do recepcionista.

Algoritmo *Recepcionista de Cinema*

Início

1 – Solicitar ao cliente o bilhete do filme.

2 – Conferir a data e o horário do filme no bilhete.

Se data/hora atual > data/hora do filme + 30 minutos **Então**

3 – Informar ao cliente que o tempo limite para entrada foi excedido.

4 – Não permitir a entrada.

Senão Se data/hora atual < data/hora do filme – 30 minutos **Então**

5 – Informar ao cliente que a sala do filme ainda não foi liberada para entrada.

6 – Não permitir a entrada.

Senão

7 – Permitir a entrada.

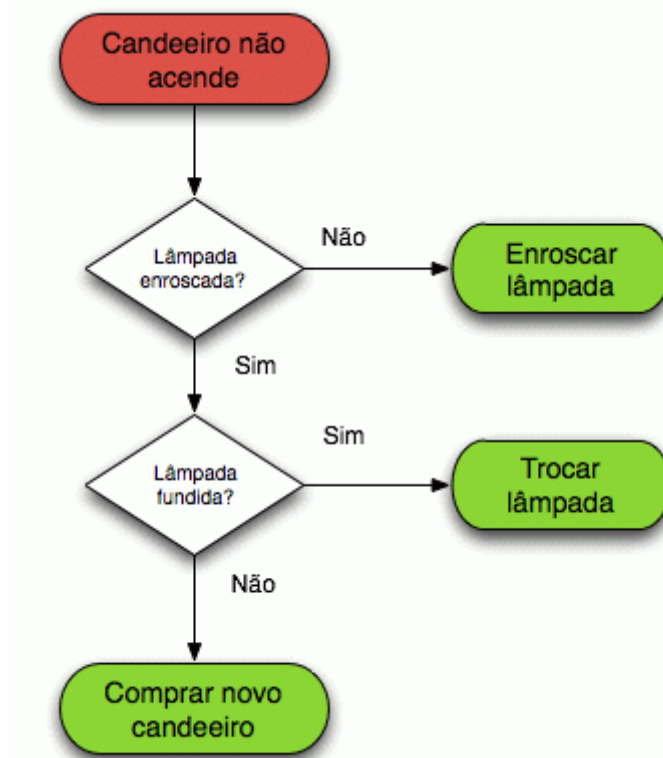
8 – Indicar ao cliente onde fica a sala do filme.

Fim-Se

Fim

Qualquer pessoa que seguir esses passos executará a função do recepcionista do cinema. Concorde? É importante notar que o algoritmo tem um fluxo que pode seguir diferentes caminhos dependendo da situação em que se encontra. Outro aspecto

interessante é que o algoritmo é finito, uma hora ele tem que acabar! Vejamos outro exemplo, dessa vez com uma representação visual: Como trocar uma lâmpada?



Fluxograma (Representação gráfica de um algoritmo)

Esta representação gráfica do algoritmo é chamada de fluxograma. Os losangos representam as decisões que são tomadas para executar um ou outro passo. Ao final, a lâmpada tem que estar funcionando.

Algoritmos na Computação

Todas as tarefas executadas pelo computador, são baseadas em algoritmos. Logo, um algoritmo deve também ser bem definido, pois é uma máquina que o executará. Uma calculadora, por exemplo, para executar a operação de multiplicação, executa um algoritmo que calcula somas até um determinado número de vezes. Abaixo, um exemplo do algoritmo de multiplicação. Para facilitar, consideremos que os fatores da multiplicação são positivos.

*Algoritmo **Multiplicação de números positivos***

Declaração de variáveis

numero1, numero2, resultado, contador: Inteiro

Início

ler(numero1)

ler(numero2)

resultado <- 0

contador <- 0

Enquanto** contador < numero2 **Faça

resultado <- resultado + numero1

```
contador <- contador + 1
Fim-Enquanto
escrever(resultado)
Fim
```

Este algoritmo pode ser considerado complexo por iniciantes, mas algoritmos deste tipo, utilizando variáveis e controle de fluxo, é muito comum em programação. Se você quer aprender programação, é necessário entendê-lo, se não conseguiu, leia-o novamente com mais atenção. Para ajudar, vamos definir algumas coisas importantes sobre o algoritmo:

- **Variável** é um espaço alocado na memória para armazenar dados. No algoritmo, foram criadas 4 variáveis.
- O símbolo "<-" representa uma atribuição de valor a uma variável. Por exemplo, (**resultado <= resultado + numero1**) atribui à variável *resultado*, o valor da própria variável *resultado*, acrescido do valor da variável *numero1*.
- O comando "**ler(numero1)**", significa que o algoritmo está lendo o que o usuário digita e armazenando na variável *numero1*.
- O comando **Enquanto** é uma estrutura de controle de fluxo do tipo "Estrutura de repetição".
- O comando **escrever(resultado)** exibe na tela o valor da variável *resultado*.

Com o tempo, a leitura e criação de algoritmos passa a ser uma coisa muito simples para um programador. Mas para isso é preciso bastante prática! Então, você pode começar fazendo um exercício, crie algoritmos para as suas tarefas do dia-a-dia a partir do momento em que você acorda. Essa é a melhor forma de aprender a criar algoritmos.