

# **Introdução à Lógica de Programação**

# O que é Lógica?

A palavra lógica está normalmente relacionada com o modo de pensar de um indivíduo em termos de racionalidade e coerência.



# O que é Lógica?

*"Podemos relacionar a lógica com a "correção do pensamento", pois uma de suas preocupações é determinar quais operações são válidas e quais não são, fazendo análises das formas e leis do pensamento. Como filosofia, ela procura saber por que pensamos assim e não de outro jeito. Com arte ou técnica, ela nos ensina a usar corretamente as leis do pensamento." Lógica de Programação – André Luiz Villar Forbellone*

*"Poderíamos dizer que a lógica é a "arte de bem pensar", que é a "ciência das formas do pensamento". Visto que a forma mais complexa do pensamento é o raciocínio, a lógica estuda a "correção do raciocínio". Podemos ainda dizer que a lógica tem em vista a "ordem da razão". Isto dá a entender que a nossa razão pode funcionar desordenadamente. Por isso a lógica estuda e ensina a colocar "ordem no pensamento". Lógica de Programação – André Luiz Villar Forbellone*

# O que é Lógica?

Todo cachorro  
é um mamífero.  
Todo mamífero  
é um animal.  
Portanto, todo  
cachorro é um  
animal.

Japão é um país  
do continente asiático.  
Todos os japoneses  
são de Japão.  
Logo, todos os  
japoneses são asiáticos.



Os exemplos ao lado  
representam um argumento  
composto por duas  
premissas e uma  
conclusão. Está sendo  
estabelecida uma relação  
que pode ser válida ou não,  
aliás, este é um dos  
objetivos da lógica -  
estudar técnicas de  
formalização, dedução e  
análise com o intuito de  
verificar a validade de  
argumentos.

# Lógica de Programação:

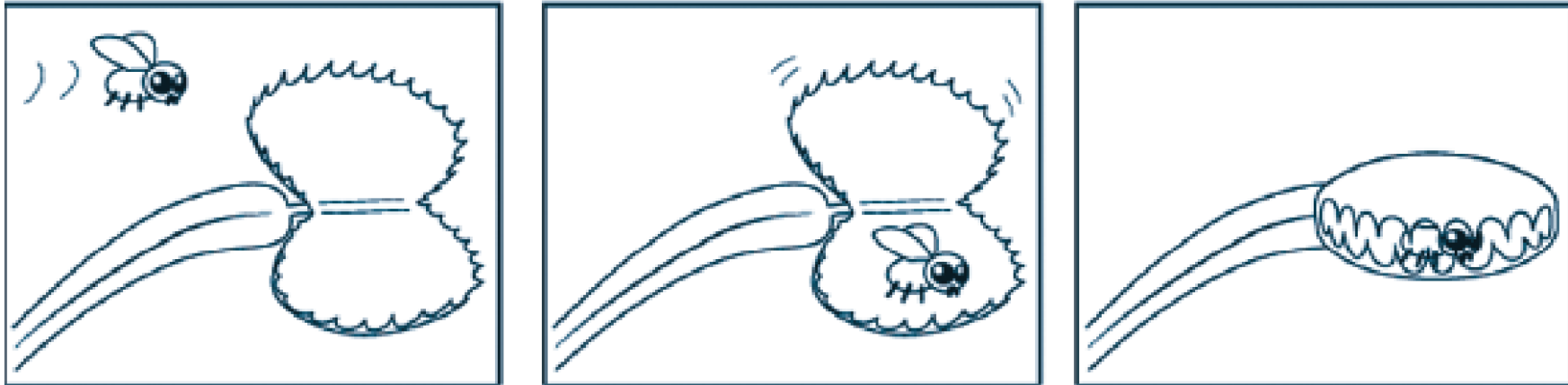
Técnica de  
*encadear*  
*pensamentos*  
para atingir  
determinado  
*objetivo*



Necessária para desenvolver programas e sistemas,  
pois permite definir a *seqüência lógica* para a  
solução de um problema

---

## Seqüência Lógica:



Estes pensamentos podem ser descritos como uma *seqüência de instruções*, que devem ser seguidas para se cumprir uma determinada tarefa

*Passos* executados até se atingir um objetivo ou solução de um problema

---

# O que são Instruções?

- Pelo dicionário Michaelis encontraremos a seguinte definição para Instrução: “Conjunto das formalidades e informações necessárias para elucidar uma causa e pô-la em estado de ser julgada.”



# O que são Instruções?

- No dicionário Michaelis especificamente a área de informática encontraremos que instrução é uma: “palavra ou expressão única que representa uma Operação.”
- 

**PROTEJA AS PASTAS  
DO  
WINDOWS**



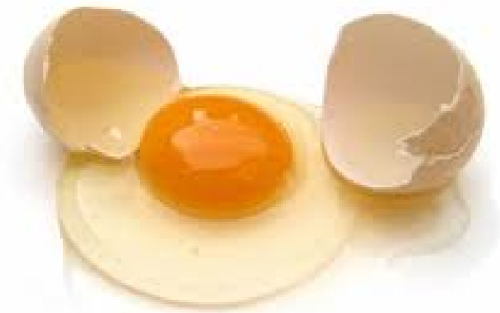
# Cuidados ao Passar Instruções

- Uma ordem isolada não permite realizar o processo completo, é necessário um conjunto de instruções colocadas em ordem seqüencial lógica.
- Claro que essas instruções têm que ser executadas em uma ordem coerente
- Uma instrução tomada isoladamente não faz sentido; para atingirmos o resultado desejado, é preciso colocar em prática o conjunto de todas as instruções, na ordem correta



EXEMPLO: para “fazer omelete”

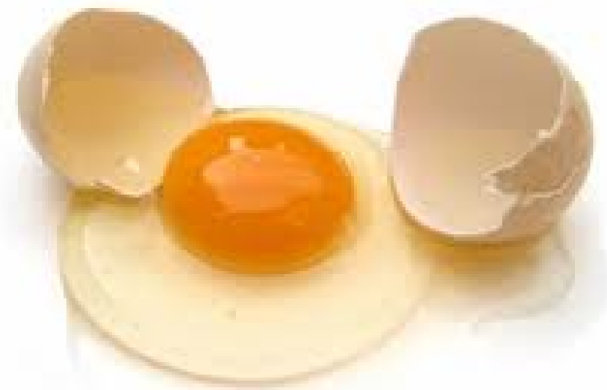
Instruções: “quebrar ovos”, “bater ovos”, “pôr sal”, “ligar fogão”, “pôr óleo na frigideira”, “pôr frigideira no fogo”, “fritar ovos batidos”, etc...



EXEMPLO: para “fazer omelete”

Quanto às instruções isoladas:

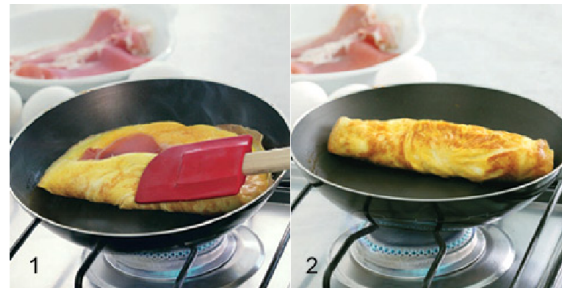
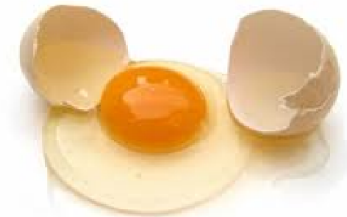
Só “quebrar ovos”, ou só “pôr óleo na frigideira”, não é suficiente para cumprir a tarefa “fazer omelete”



EXEMPLO: para “fazer omelete”

Quanto à seqüência lógica:

Se executarmos “fritar ovos batidos”  
antes de “bater ovos”, ou pior,  
antes de “quebrar ovos”, não  
iremos cumprir a tarefa “fazer  
omelete”



# O que são Instruções?

Cada um dos *passos*, cada uma das ações a tomar (obedecendo a *seqüência lógica*) para ir resolvendo o problema, ou para ir executando a tarefa

Em informática, é a informação que indica a um computador uma *operação elementar* a executar

Ex.: “somar”, “subtrair”, “comparar se é maior”, etc

Uma só instrução não resolve problemas

Executar um *conjunto de instruções*

Executar em uma *seqüência lógica*

---

# O que são Instruções?

Cada um dos *passos*, cada uma das ações a tomar (obedecendo a *seqüência lógica*) para ir resolvendo o problema, ou para ir executando a tarefa

Em informática, é a informação que indica a um computador uma *operação elementar* a executar

Ex.: “somar”, “subtrair”, “comparar se é maior”, etc

Uma só instrução não resolve problemas

Executar um *conjunto de instruções*

Executar em uma *seqüência lógica*

---

# O que são Instruções?

Cada um dos *passos*, cada uma das ações a tomar (obedecendo a *seqüência lógica*) para ir resolvendo o problema, ou para ir executando a tarefa

Em informática, é a informação que indica a um computador uma *operação elementar* a executar

Ex.: “somar”, “subtrair”, “comparar se é maior”, etc

Uma só instrução não resolve problemas

Executar um *conjunto de instruções*

Executar em uma *seqüência lógica*

---

# O que são Instruções?

Cada um dos *passos*, cada uma das ações a tomar (obedecendo a *seqüência lógica*) para ir resolvendo o problema, ou para ir executando a tarefa

Em informática, é a informação que indica a um computador uma *operação elementar* a executar

Ex.: “somar”, “subtrair”, “comparar se é maior”, etc

Uma só instrução não resolve problemas

Executar um *conjunto de instruções*

Executar em uma *seqüência lógica*

---

# O que são Instruções?



# O que são Instruções?

Cada um dos *passos*, cada uma das ações a tomar (obedecendo a *seqüência lógica*) para ir resolvendo o problema, ou para ir executando a tarefa

Em informática, é a informação que indica a um computador uma *operação elementar* a executar

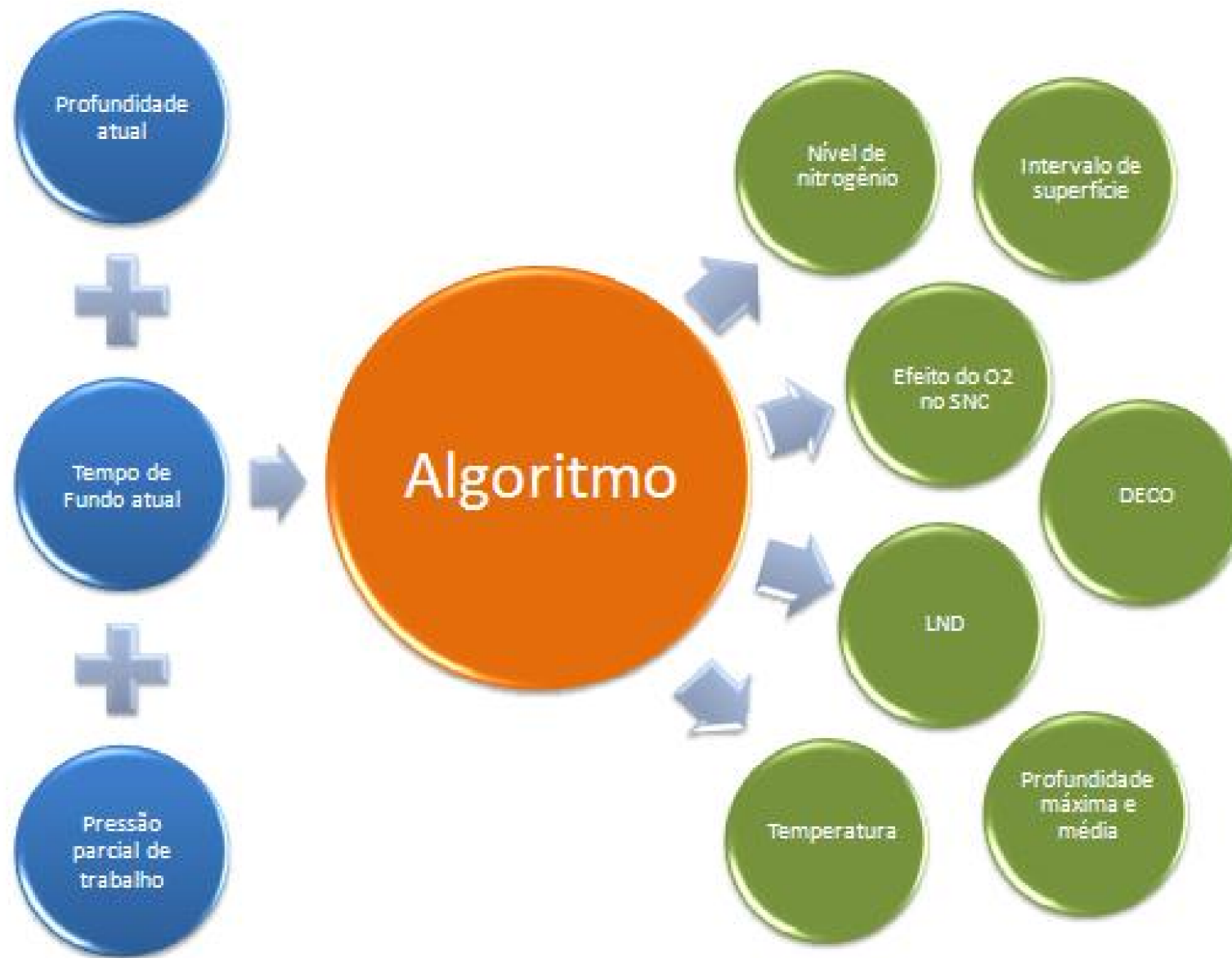
Ex.: “somar”, “subtrair”, “comparar se é maior”, etc

Uma só instrução não resolve problemas

Executar um *conjunto de instruções*

Executar em uma *seqüência lógica*

---

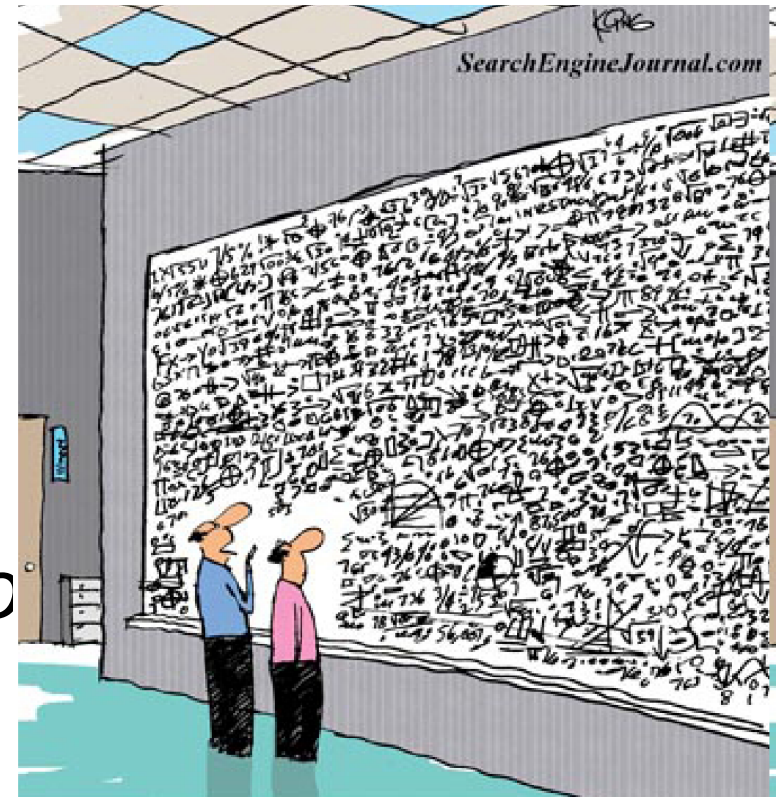


# O que é o Algoritmo?

*“Um conjunto finito de regras que provê uma seqüência de operações para resolver um tipo de problema específico”*

*(KNUTH)*

*“Seqüência ordenada, e não ambígua, de passos que levam à solução de um dado problema”*  
*(TREMBLAY)*



... E simplificando, é desta forma que iremos  
melhorar nosso ranking nos  
sites de buscas.

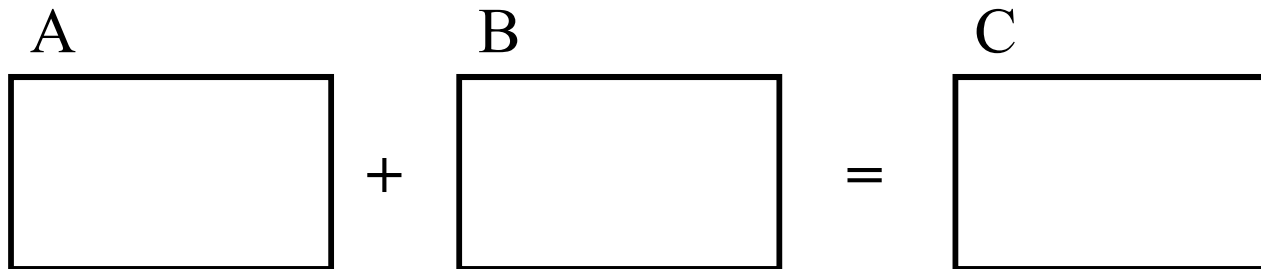
# O que é o Algoritmo?

*“Processo de cálculo, ou de resolução de um grupo de problemas semelhantes, em que se estipulam, com generalidade e sem restrições, as regras formais para a obtenção do resultado ou da solução do problema”  
(AURÉLIO)*



# O que é o Algoritmo?

- **Seqüência finita de passos** que levam à execução de uma tarefa
- Claro e preciso. Ex. “somar dois números”:
  - Escrever primeiro número no retângulo A
  - Escrever segundo número no retângulo B
  - Somar o número do retângulo A com o número do retângulo B e escrever o resultado no retângulo C



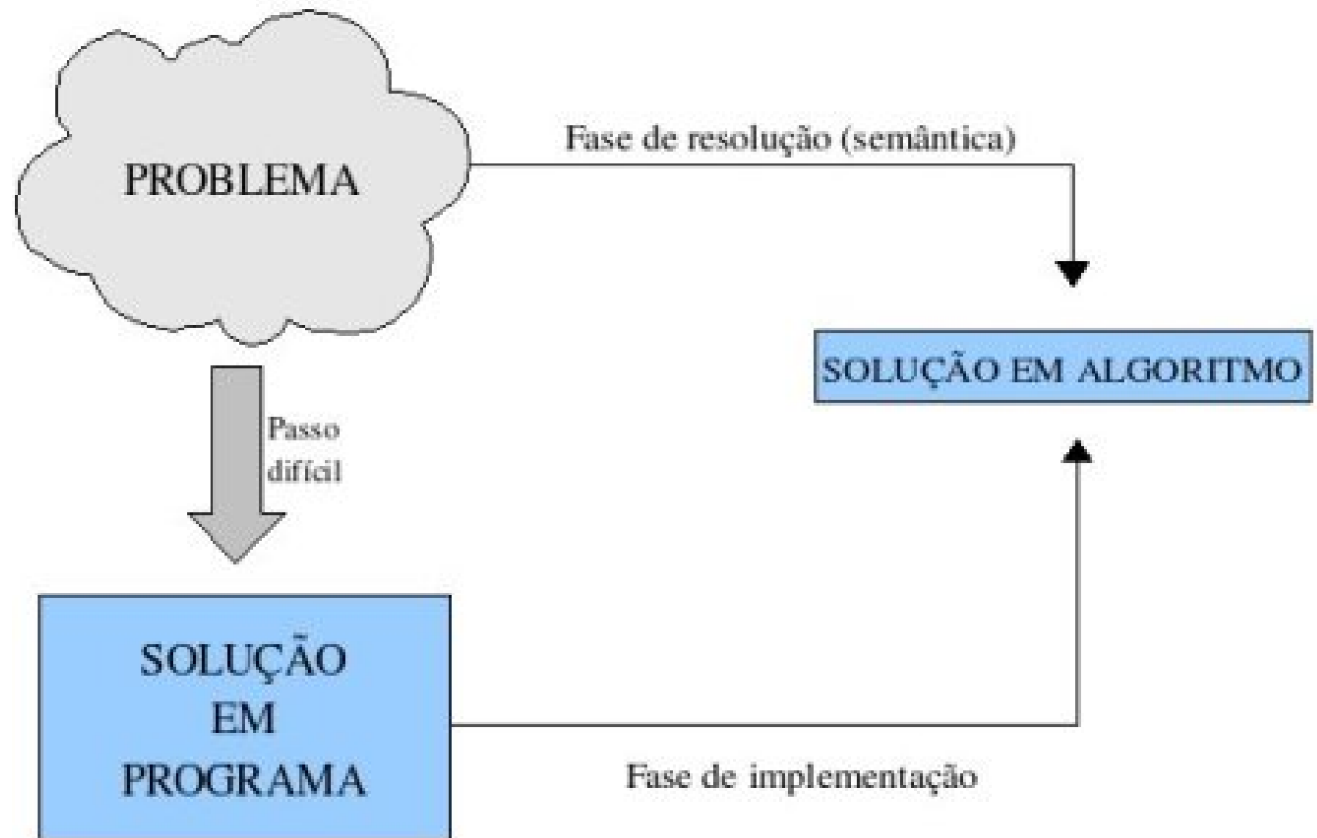
# O que é o Algoritmo?

Os passos devem ser especificados com clareza e exatidão e devem ser realizados em cada uma das fases do processo a ser automatizado, bem como a seqüência em que estas fases devem ser realizadas. A especificação da seqüência ordenada de passos que deve ser seguida para a realização de uma tarefa, garantindo a sua repetibilidade, dá-se o nome de algoritmo.



# Para que serve os Algoritmo?

*Serve como modelo para programas, pois sua linguagem é intermediária à linguagem humana e às linguagens de programação, sendo então, uma boa ferramenta na validação da lógica de tarefas a serem automatizadas.*



# Características de um Algoritmo

- Todo algoritmo deve apresentar algumas características básicas:

- Ter um início;
- Ter um fim;
- Não dar margem à dupla interpretação (não ter duplo sentido);
- Ter a capacidade de receber dado(s) de entrada do mundo exterior;
- Poder gerar informações de saída para o mundo externo ao do ambiente do algoritmo;

- Ser efetivo (todas as etapas especificadas no algoritmo devem ser alcançáveis em um tempo finito).

## Algoritmo "Trabalhar pela manhã"

1. Acordar
2. Tomar banho
3. Vestir-se
4. Tomar café
5. Tirar o carro da garagem
6. Ir para o trabalho

## Algoritmo "Tomar banho"

1. Tirar a roupa
2. Ligar Chuveiro
3. Molhar o corpo
4. Ensaboar-se
5. Enxaguar o corpo
6. Desligar chuveiro
7. Vestir-se

# Formas de Representar um Algoritmo

- *Descrição narrativa:*
    - o algoritmo é descrito, passo a passo, e todas as tarefas a serem executadas na língua mãe, no nosso caso em português.
    - os algoritmos são expressos diretamente em **linguagem natural**.
-

# Formas de Representar um Algoritmo

- *Fluxograma*
  - A representação através de símbolos gráficos que mostram a seqüência de execução é uma das maneiras possíveis de se representar os algoritmos. Esta representação é chamada de fluxograma. Existem símbolos padronizados para início, entrada de dados, cálculos, saída de dados, fim e outras funções.





# Verdade sobre os Algoritmos

*Algoritmos não se aprendem:*

- Copiando algoritmos*
- Estudando algoritmos*

*Algoritmos só se aprendem:*

- Construindo algoritmos*
  - Testando algoritmos*
-

# Verdade sobre os Algoritmos

*Algoritmos não se aprendem:*

- Copiando algoritmos*
- Estudando algoritmos*

*Algoritmos só se aprendem:*

- Construindo algoritmos*
  - Testando algoritmos*
-

# O que são Programas de Computador?

- Uma coleção de instruções que descrevem uma tarefa a ser realizada por um computador. O termo pode ser uma referência ao código fonte, escrito em alguma linguagem de programação, ou ao arquivo que contém a forma executável deste código fonte.

(Fonte: Wikipédia)

