

# Algoritmos

**Prof. Jonatas Bastos**

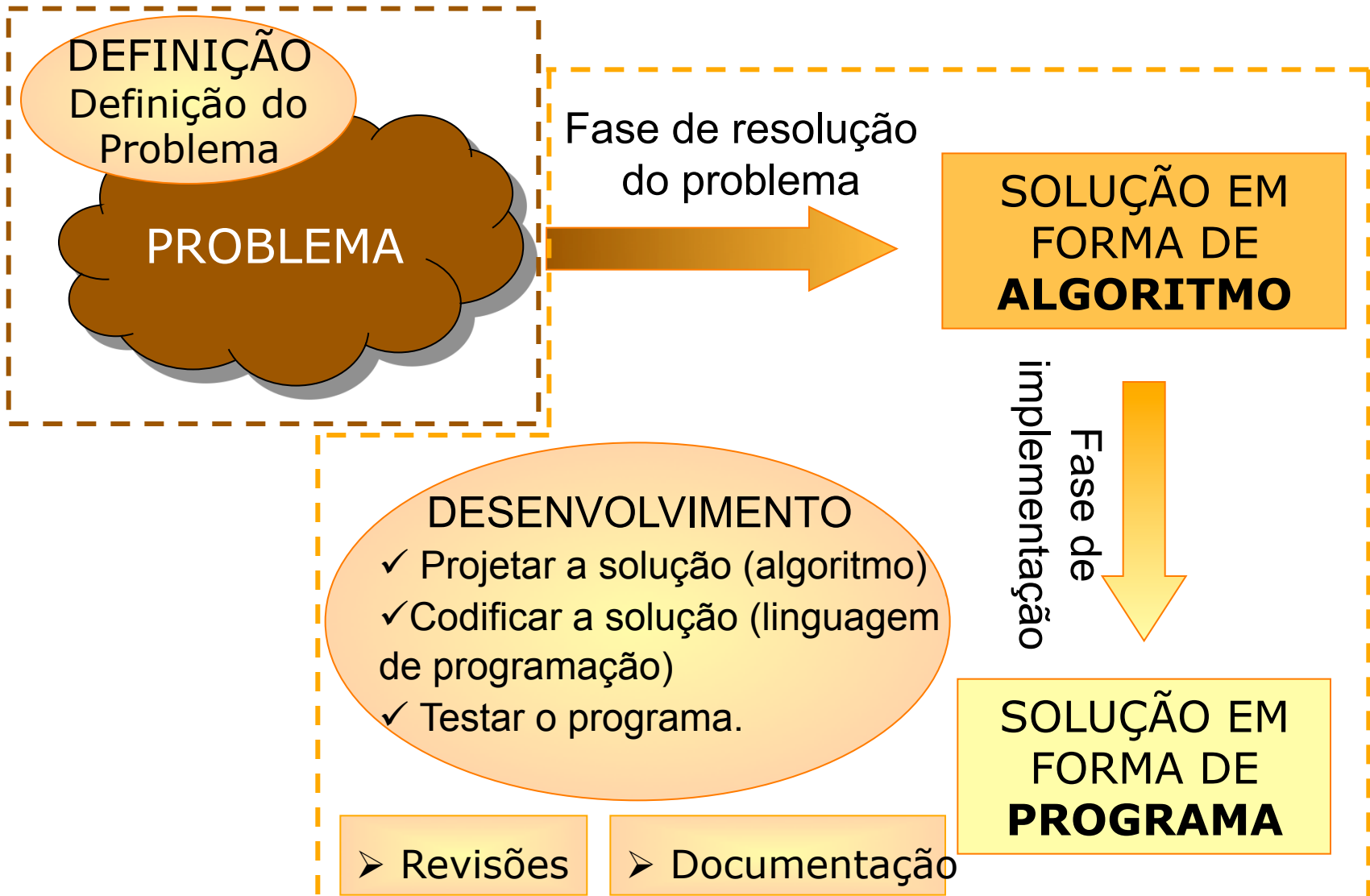
**Email: [jonatasfbastos@gmail.com](mailto:jonatasfbastos@gmail.com)**

**Site: <http://jonatasfbastos.wordpress.com/>**

# Dúvidas!

- ❑ Como os computadores entendem os algoritmos?
- ❑ Os computadores não entendem os passos descritos para resolver um problema.
- ❑ É necessário traduzir os passos em uma linguagem que o computador entenda.

# Etapas da Construção de Programas



# Programação e Níveis de Linguagem

- Os programas criados por programadores são escritos em Linguagens de Programação de **alto nível**.
- Linguagens de programação de alto nível estão próximas aos programadores e portanto são mais claras e compreensíveis do que as linguagens de baixo nível.

Exemplo:

```
SOMA := X + 5;
```

Python

Java



begin  
if  
end  
else  
case

Ruby

# Linguagem de Máquina

- Cada CPU possui um conjunto de instruções expressas em uma linguagem (a única) que ela pode entender;
- Constituída inteiramente de números binários
  - Muito difícil para um humano entender e manipular !!

Exemplo:

código da operação	operando
001	01010



Programadores de verdade programam em binário !



- ❑ Linguagem de baixo nível próxima à linguagem de máquina.
- ❑ Possui a mesma estrutura e conjunto de instruções que a linguagem de máquina, porém
  - permite que o programador utilize nomes (chamados **mnemônicos**) e símbolos em lugar dos números.

## Exemplo:

código da operação	operandos	significado
MOV	EAX, 4	Move 4 para o registrador EAX
ADD	EAX, 5	Soma 5 ao registrador EAX

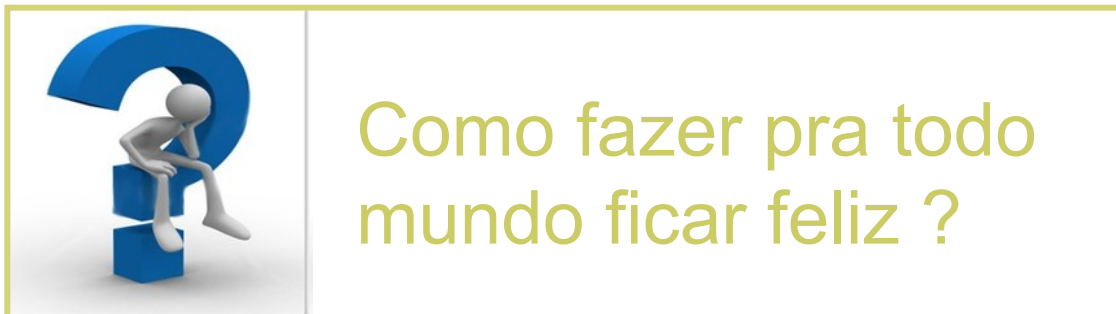
# Ling. Alto Nível → Ling. Máquina



Programadores preferem  
**Linguagem de alto nível**



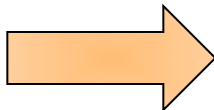
Computadores só entendem  
**Linguagem de máquina**



# Ling. Alto Nível Ling. Máquina



CÓDIGO  
FONTE



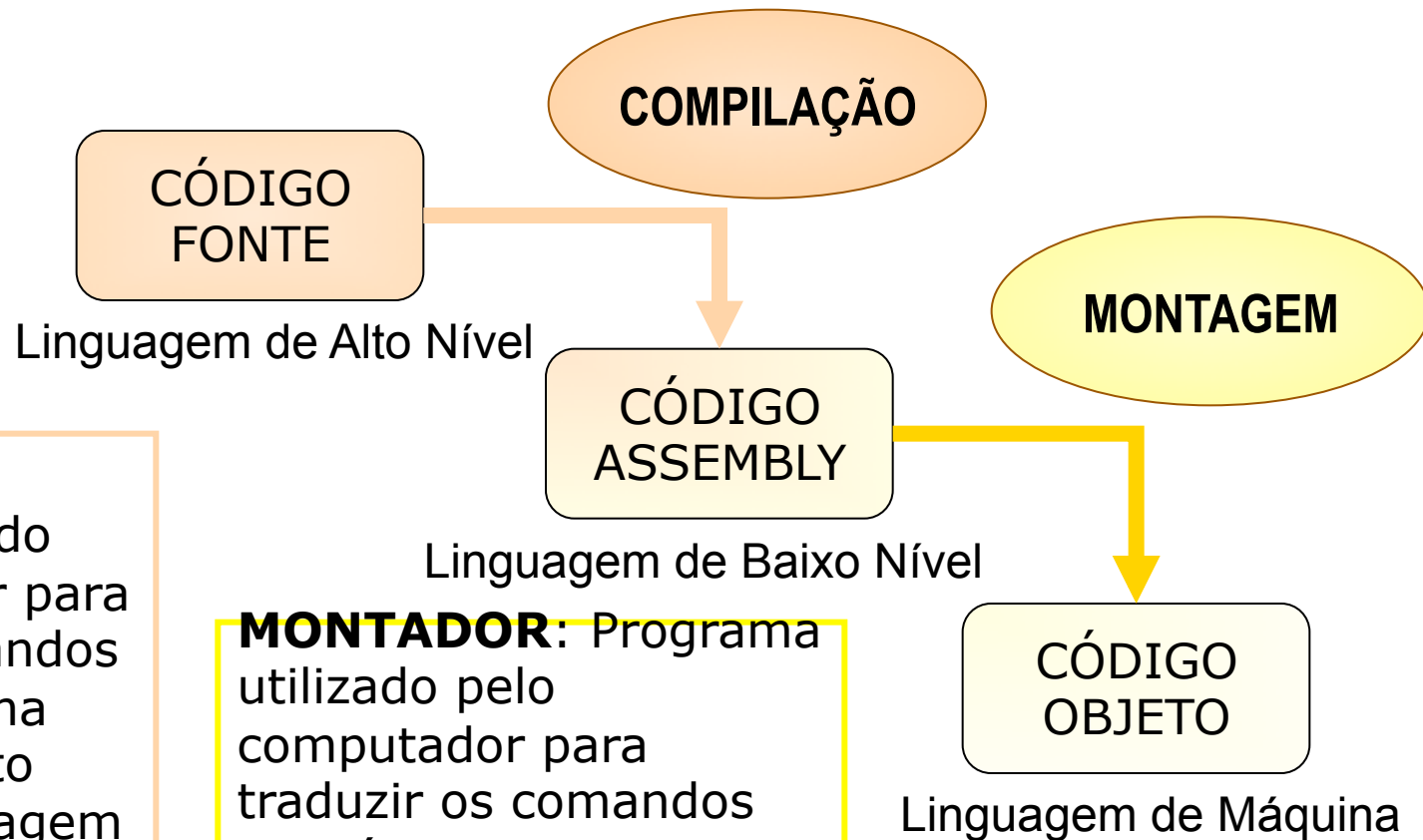
**TRADUÇÃO**



CÓDIGO  
OBJETO



# Processo de Tradução (1)

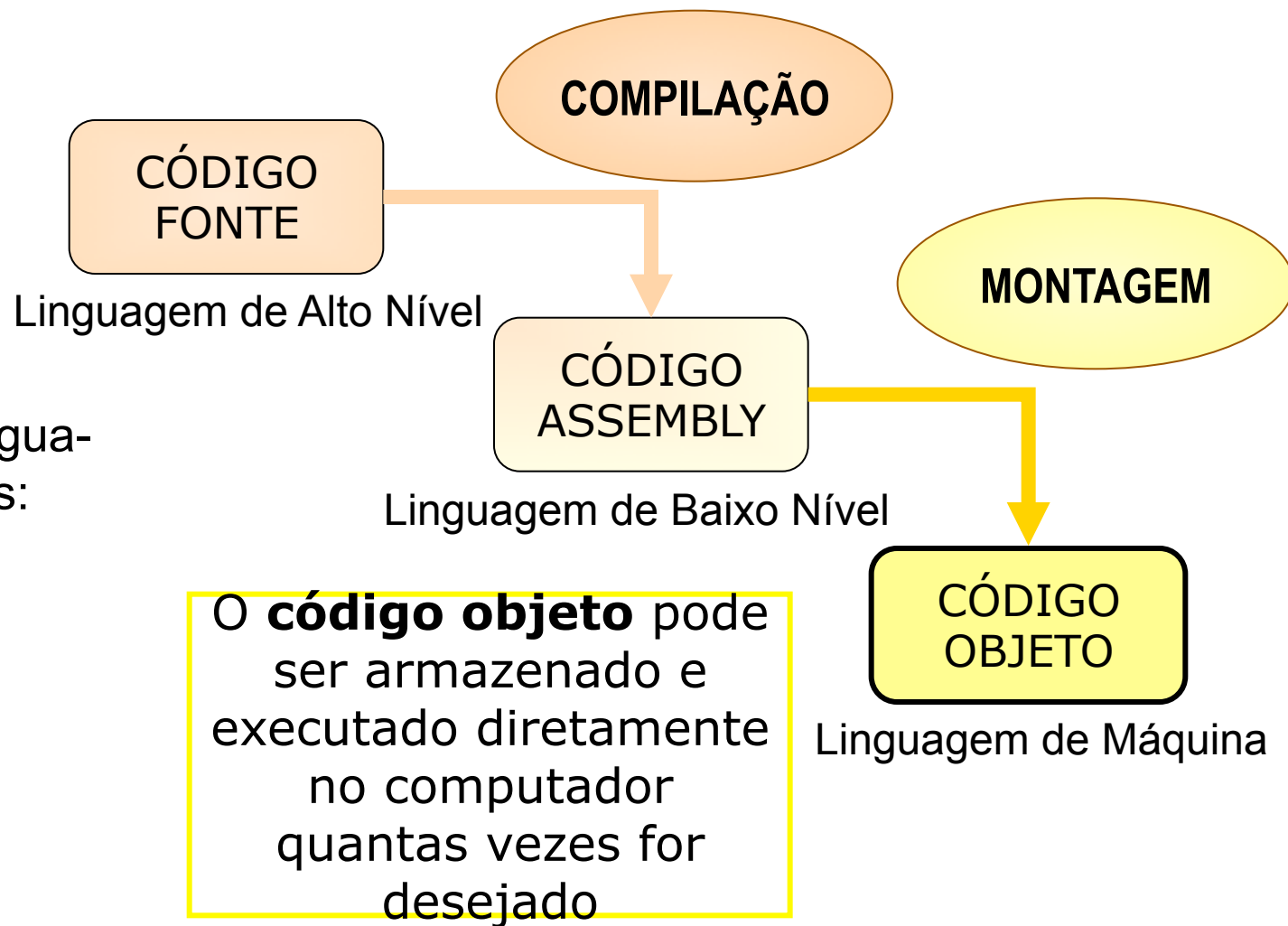


## **COMPILADOR:**

Programa utilizado pelo computador para traduzir os comandos simbólicos de uma linguagem de alto nível, para linguagem de nível mais baixo (possivelmente, de máquina).

**MONTADOR:** Programa utilizado pelo computador para traduzir os comandos simbólicos de uma linguagem de baixo nível, para linguagem de máquina.

# Processo de Tradução (1)



Exemplos de linguagens compiladas:

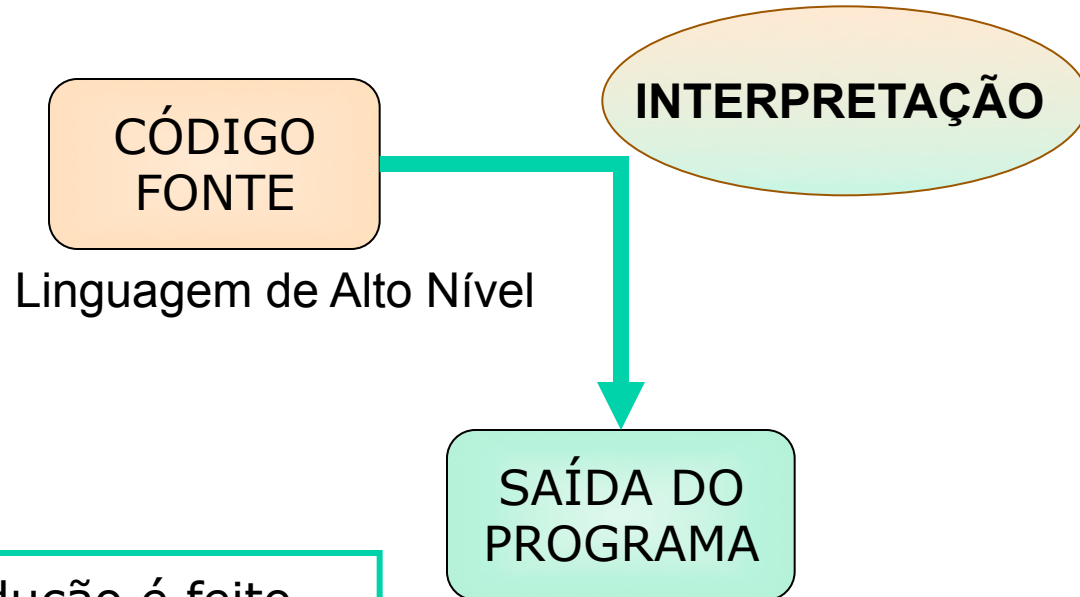
- Pascal
- C
- C++
- Cobol

# Processo de Tradução (2)

## INTERPRETADO

**R:** Programa que lê, traduz e executa uma declaração do programa por vez.

- Todo processo de tradução é feito em tempo de execução.
- Não é produzido um código objeto do programa completo que possa ser armazenado e executado posteriormente.

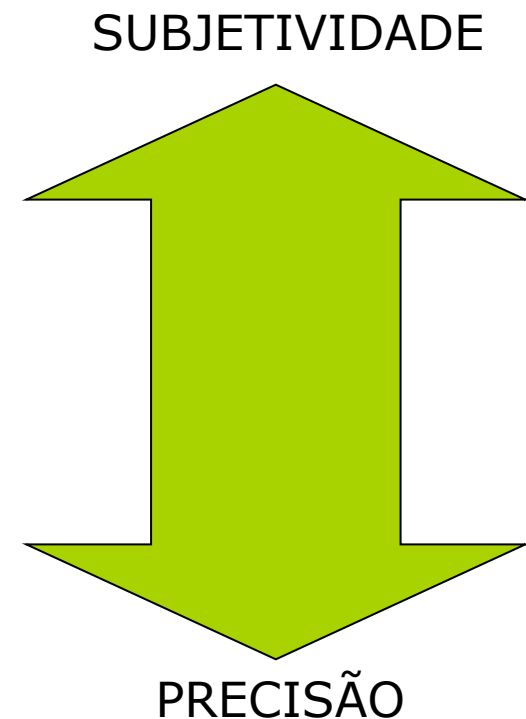


Exemplos de linguagens interpretadas:

- Javascript;
- Python;
- Perl, Basic.

# Representações de um Algoritmo

- Descrição narrativa;
- Fluxograma
- Pseudocódigo,  
Português Estruturado  
ou Portugol;



# Representações de um Algoritmo

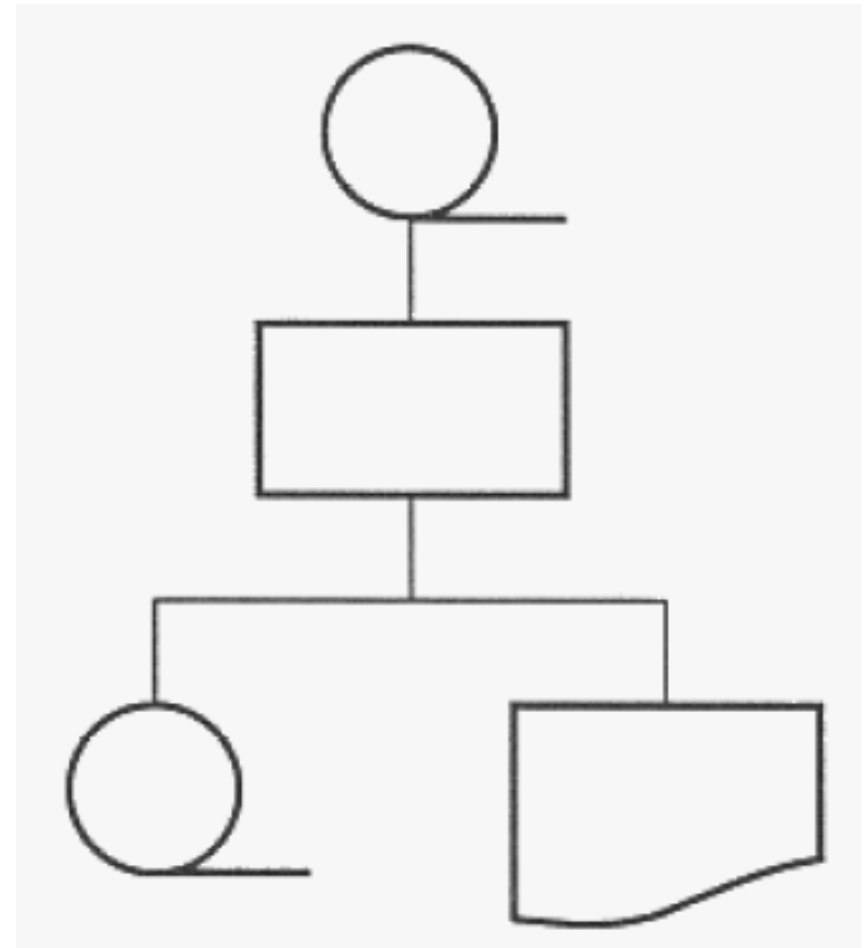
- Descrição Narrativa:
  - Preferencialmente um verbo por frase;
  - Frases curtas e simples;
  - Objetividade;
  - Evitar palavras com sentido dúbio;

- Descrição Narrativa:
  - Dobro de um número (Dobro = Número  $\times$  2):
    - Digitar um número;
    - Gravar em uma variável;
    - Multiplicar o número digitado por 2;
    - Gravar o resultado em outra variável;
    - Mostrar o resultado da operação.

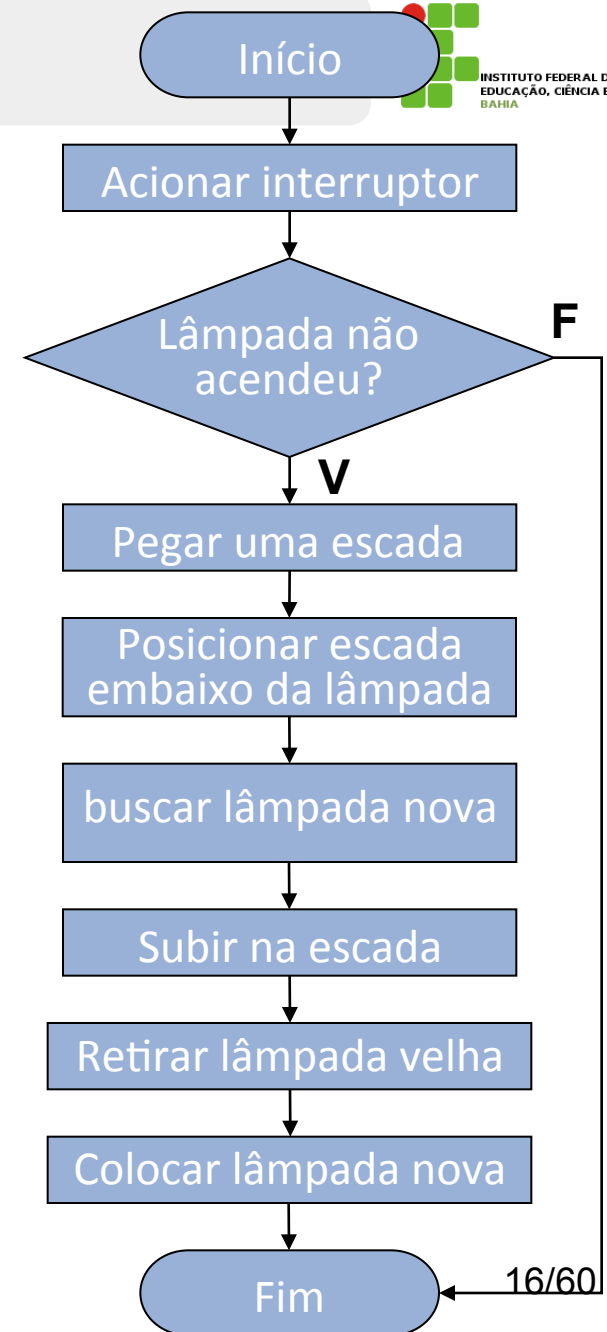
# Representações de um algoritmo

## □ Fluxogramas

- Descrevem o fluxo de ação de um determinado trabalho lógico, seja manual ou mecânico;
- Usa símbolos convencionais, permitindo poucas variações;
- Representado por símbolos geométricos;



# Fluxograma – Algoritmo troca de lâmpada





# Representações de um algoritmo

- Português estruturado (Portugol)
  - Linguagem mais restrita que o português em linguagem natural;
  - Simplificação extrema do português;
  - Significados bem definidos para todos os termos utilizados nas instruções;
  - Possui um conjunto de palavras e regras específicas (sintaxe da linguagem);
  - Representação utilizada no curso;

# Pseudocódigo – Algoritmo troca de lâmpada

Início

```
Ir até o interruptor do primeiro soquete;  
enquanto <houver soquetes> faça  
    acionar interruptor;  
    se <lâmpada não acender> então  
        pegar uma escada;  
        posicionar a escada embaixo da lâmpada;  
        buscar uma lâmpada nova;  
        subir na escada;  
        retirar a lâmpada velha;  
        colocar a lâmpada nova;  
        ir até interruptor do próximo soquete;  
    fim se  
fim enquanto
```

Fim

# Exemplo de Algoritmo

*{ declaração de variáveis }*

**real:** p1, p2, p3, p4, Media;

**início**

*{ comandos de entrada de dados }*

leia (p1);

leia (p2);

leia (p3);

leia (p4);

*{ processamento - Calculo da media }*

Media  $\leftarrow$  (p1 + p2 + p3 + p4)/4;

*{ saída de dados }*

escreva (' Media final = ', Media);

**fim.**