

Programação de Computadores



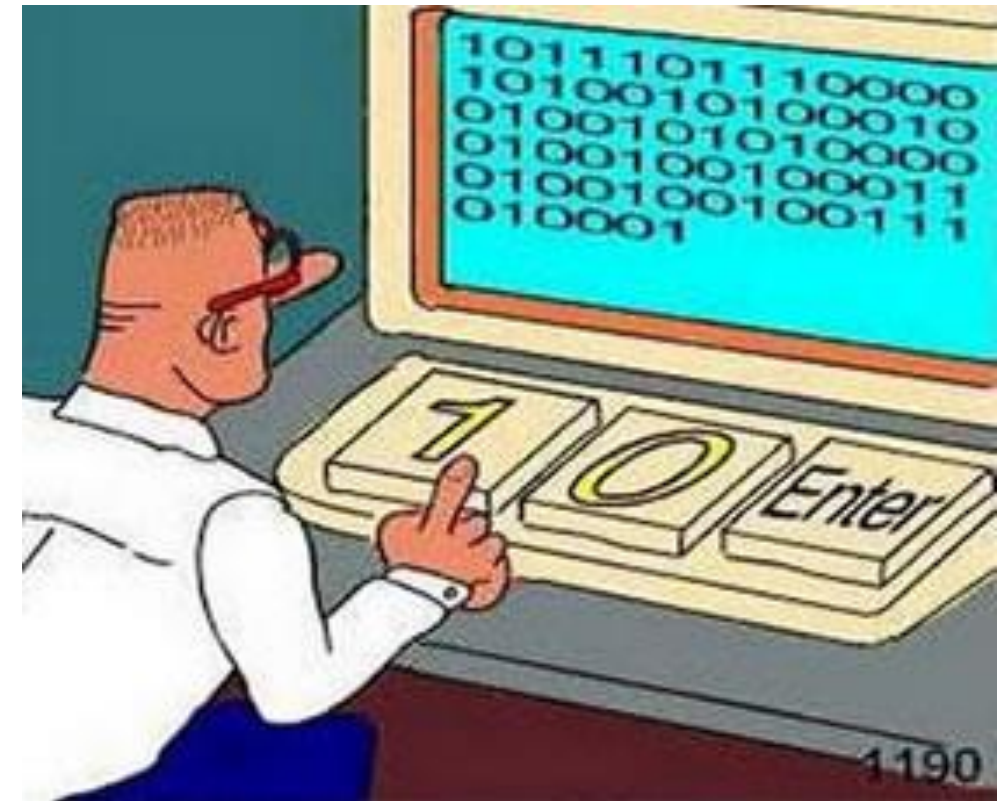
- ✓ Conceitos iniciais

- ✔ A programação de computadores é uma atividade que leva à representação dos passos necessários à resolução de um problema em linguagem de programação.
- ✔ Para dar início ao aprendizado dessa atividade, é importante compreender seu contexto, seu propósito, os conceitos básicos subjacentes, bem como tomar contato com as ferramentas necessárias para sua realização.



As instruções estão escritas em uma linguagem que permite a comunicação entre o programador e o computador (0's e 1's) →

Linguagem de programação

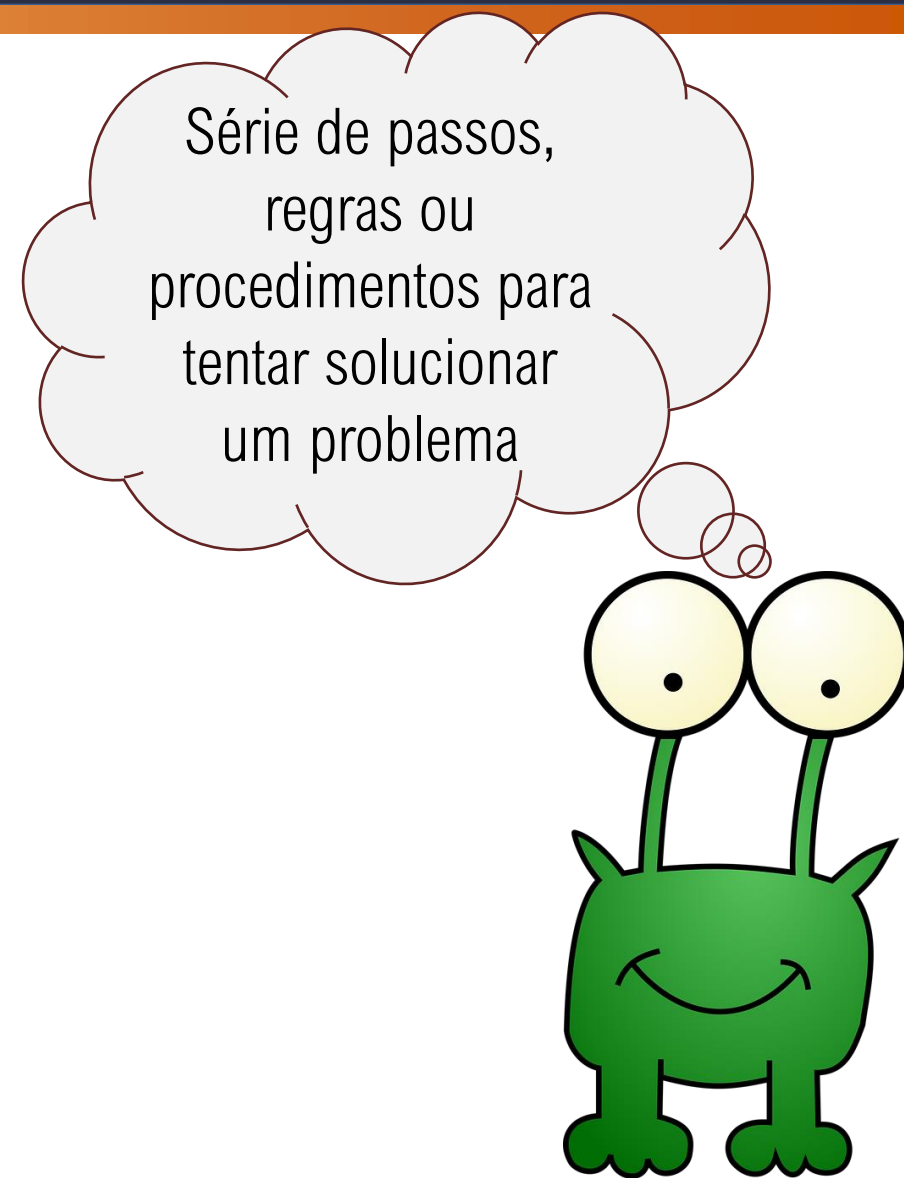




De forma bem genérica podemos dizer que construir um programa envolve as seguintes etapas:

- ✓ Analisar o problema apresentado em busca de uma solução;
- ✓ Escrever esta solução na forma de um algoritmo;
- ✓ Implementar este algoritmo na linguagem de programação escolhida;
- ✓ Realizar testes;
- ✓ Corrigir erros, tanto sintáticos como lógicos;
- ✓ Gerar pacote de instalação.

- ✓ Em computação pode ser definido como uma sequência de instruções ou operações básicas, cuja execução, em tempo finito resolve um problema computacional.
- ✓ Ele pode ser representado graficamente, em pseudocódigo ou diretamente na linguagem escolhida para desenvolvimento do programa.
- ✓ A partir do algoritmo será construído um programa, que estará escrito em alguma linguagem de programação para que possa ser executado em um computador.



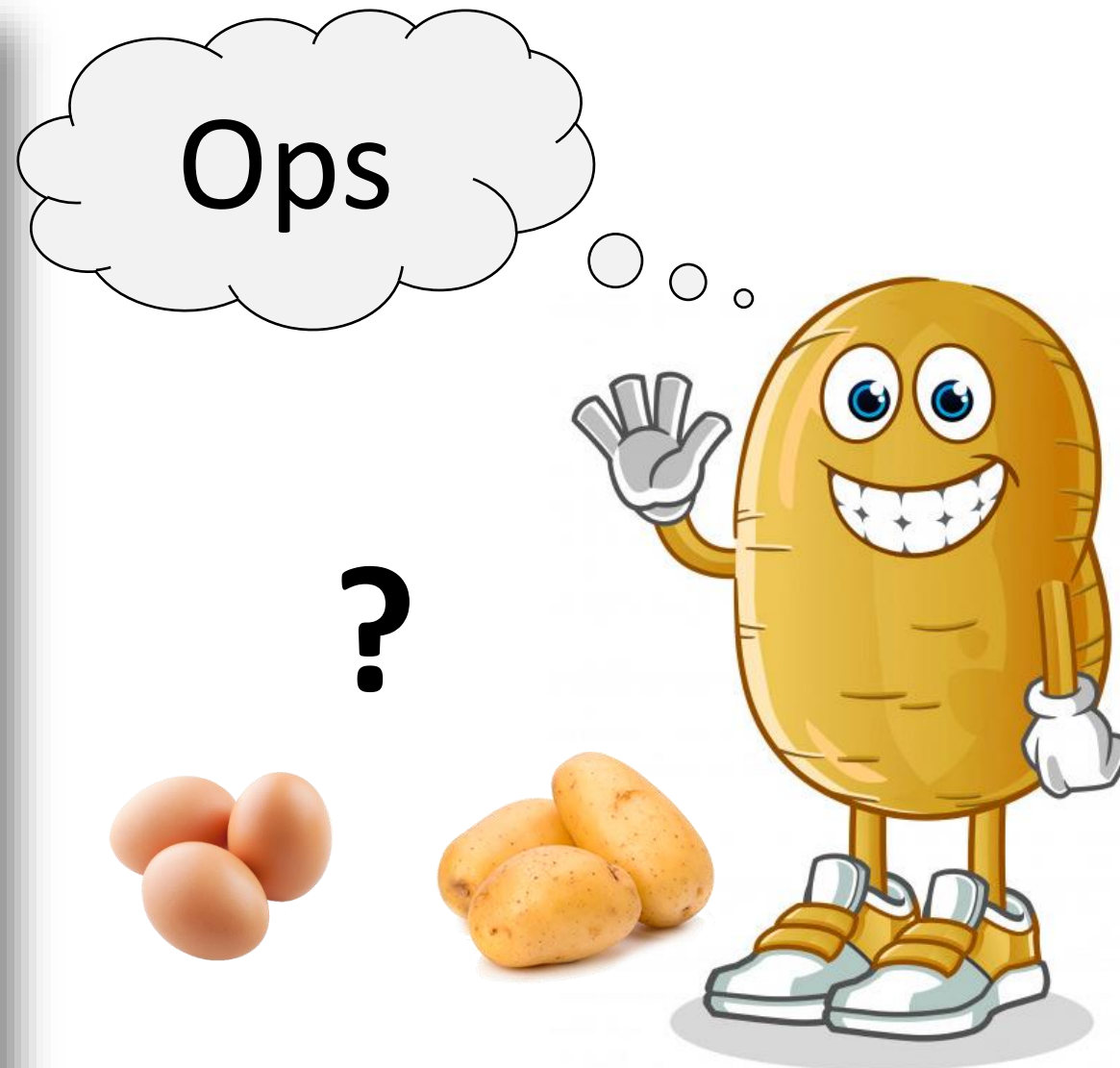
ALGORITMO 1.1 Troca de lâmpada

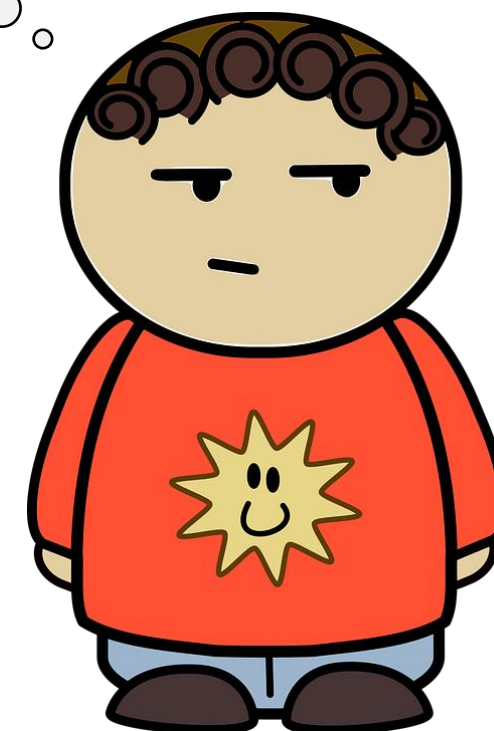
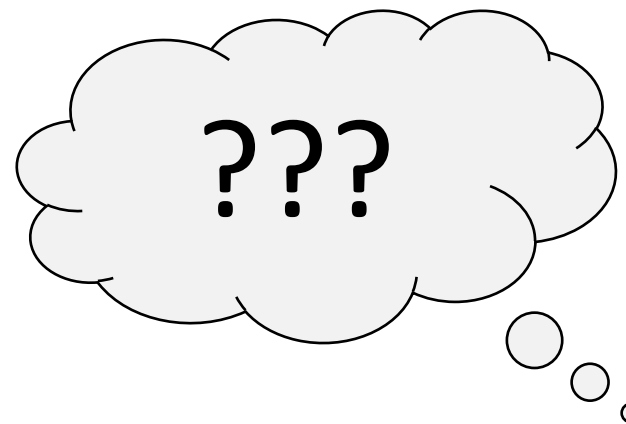
- pegar uma escada;
 - posicionar a escada embaixo da lâmpada;
 - buscar uma lâmpada nova;
 - subir na escada;
 - retirar a lâmpada velha;
 - colocar a lâmpada nova.
-



- descer da escada
- guardar a escada

muitas vezes
podemos
aprimorar um
algoritmo...





Você sabe dizer como um algoritmo influencia sua vida?

A estrutura de um algoritmo em português coloquial

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



Você sabe dizer como um algoritmo influencia sua vida?

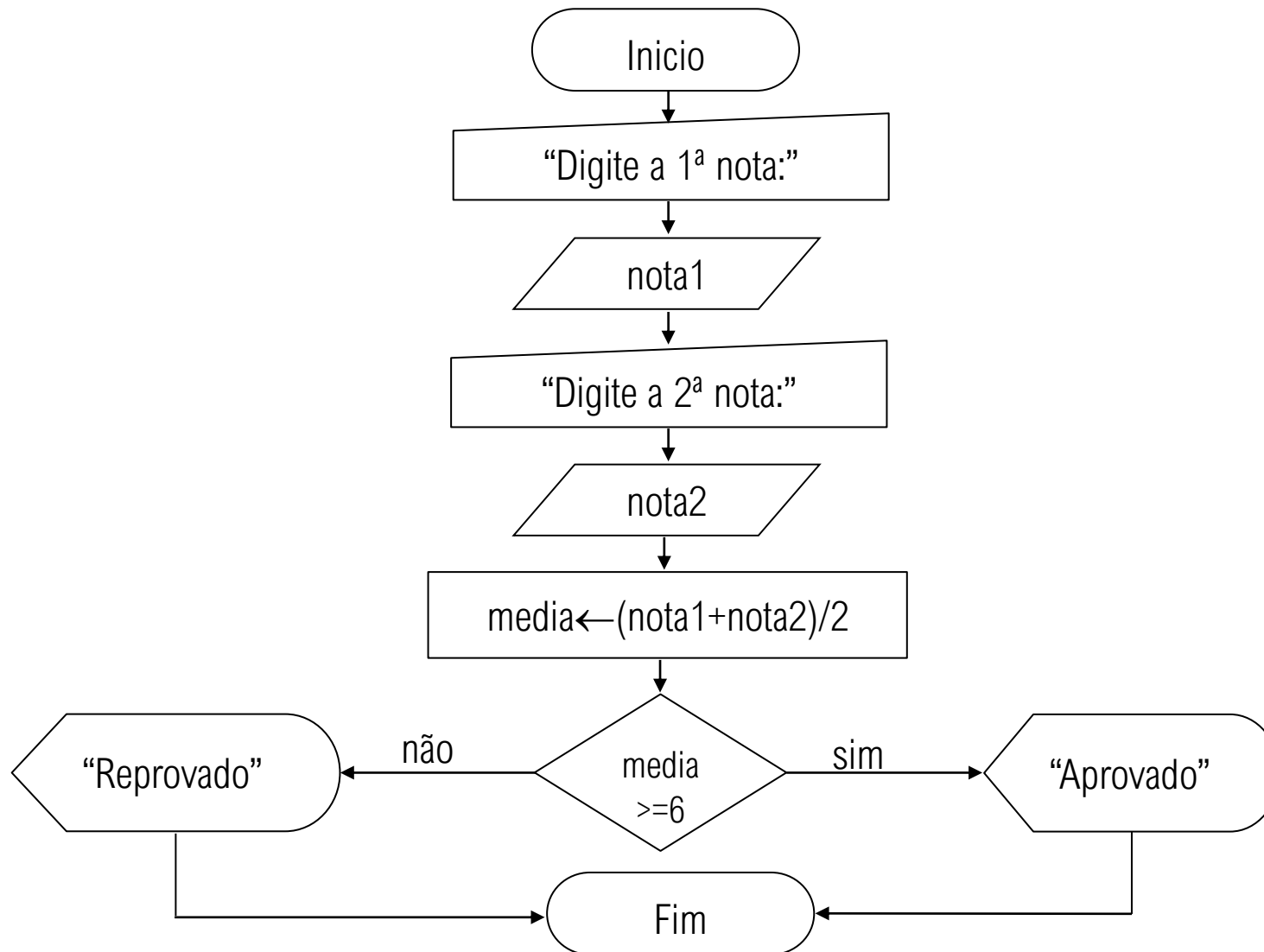
Algoritmo(receita de bolo):

- 1) Bater duas claras em castelo;
- 2) Adicionar duas gemas;
- 3) Adicionar um xícara de açúcar;
- 4) Adicionar duas colheres de manteiga;
- 5) Adicionar uma xícara de leite de coco;
- 6) Adicionar farinha e fermento;
- 7) Colocar numa forma e levar ao forno em lume brando.



```
algoritmo exemplo
    real media, nota1, nota2
inicio
    escreva ("Digite a 1ª nota: ")
    leia (nota1)
    escreva ("Digite a 2ª nota: ")
    leia (nota2)
    media = (nota1 + nota2) / 2
    se (media >= 6) {
        escreva ("Aprovado")
    } senão {
        escreva ("Reprovado")
    }
fim
```



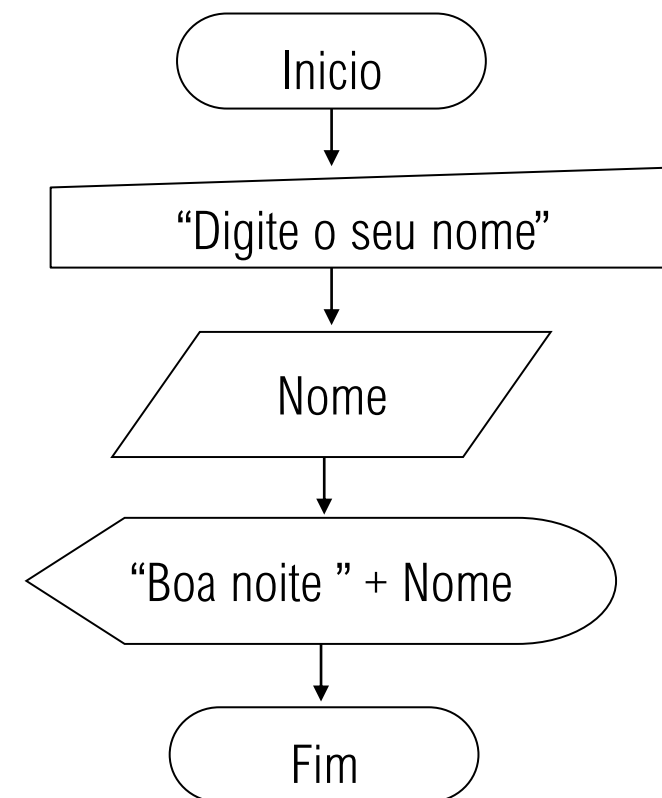


- ✔ Porque prepara uma lógica adequada, correta, para resolver um determinado problema.
- ✔ A partir dele será construído um programa, que estará escrito em alguma linguagem de programação para que possa ser executado em um computador.
- ✔ Pode ser programado em diferentes linguagens de programação.
- ✔ É imprescindível considerar todas as operações ou passos necessários de um algoritmo e a ordem em que deverão ser executadas estas operações pelo computador.



Queremos fazer um algoritmo que o usuário entre com o seu nome e o sistema imprima na tela “Boa noite “ e o nome dessa pessoa.

```
algoritmo boaNoite  
  literal nome  
  inicio  
    escreva("Digite o seu nome: ")  
    leia(nome)  
    escreva("Boa noite " + nome)  
  fim
```



Para que o computador “entenda” um programa escrito em uma linguagem (de alto nível) é necessário um meio de tradução entre a linguagem de alto nível utilizada no programa e a linguagem de máquina.



```
1 import javax.swing.*;
2 public class Teste {
3     public static void main(String args[ ]) {
4         String nome;
5         nome = JOptionPane.showInputDialog("Digite o seu nome");
6         JOptionPane.showMessageDialog(null, "Boa noite " + nome);
7     }
8 }
```



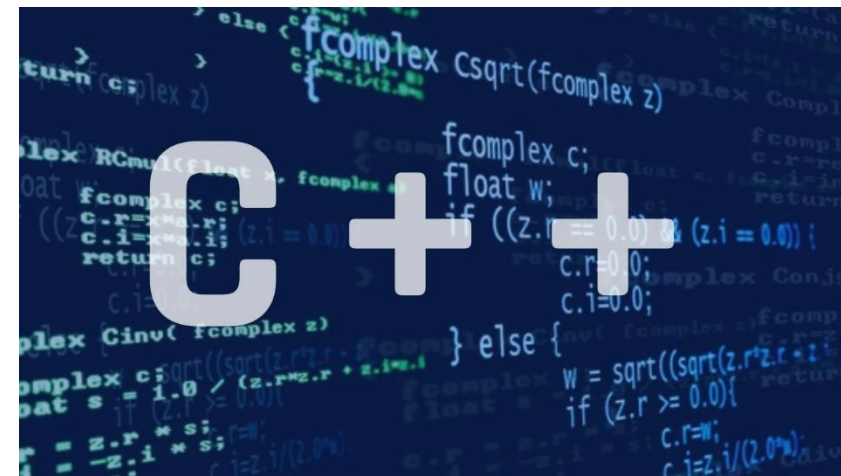
```
#include<stdio.h>

void main()
{
    char nome[30];
    printf("Digite o seu nome: ");
    gets(nome);
    printf("Boa Noite %s", nome);
}
```




```
#include<iostream>
using namespace std;

void main()
{
    char nome[30];
    cout << "Digite o seu nome: ";
    gets(nome);
    cout << "Boa Noite" << nome;
}
```



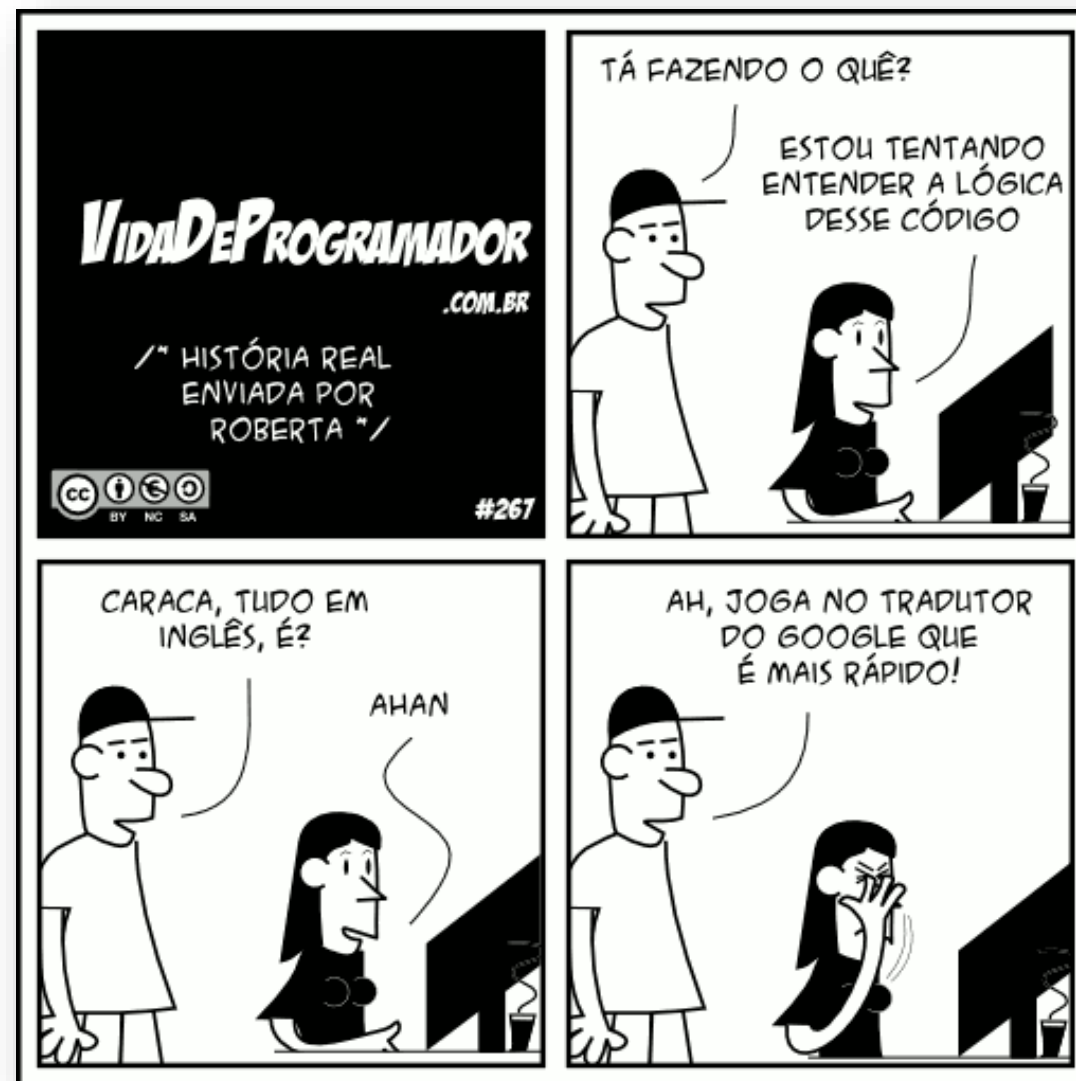
```
1 <script>
2     let nome;
3     nome = prompt("Digite o seu nome");
4     alert("Boa noite " + nome);
5 </script>
```

A yellow square containing the letters 'JS' in a bold, black, sans-serif font, representing the JavaScript logo.

Para que o computador “entenda” um programa escrito em uma linguagem (de alto nível) é necessário um meio de tradução entre a linguagem de alto nível utilizada no programa e a linguagem de máquina.

Para essa tarefa temos basicamente dois métodos:

- ✓ Compilador
- ✓ Interpretador



- ✓ Traduz e faz a checagem da sintaxe e envia para execução, instrução por instrução.
- ✓ Precisa estar presente todas as vezes que vamos executar o programa e o processo acima é repetido.

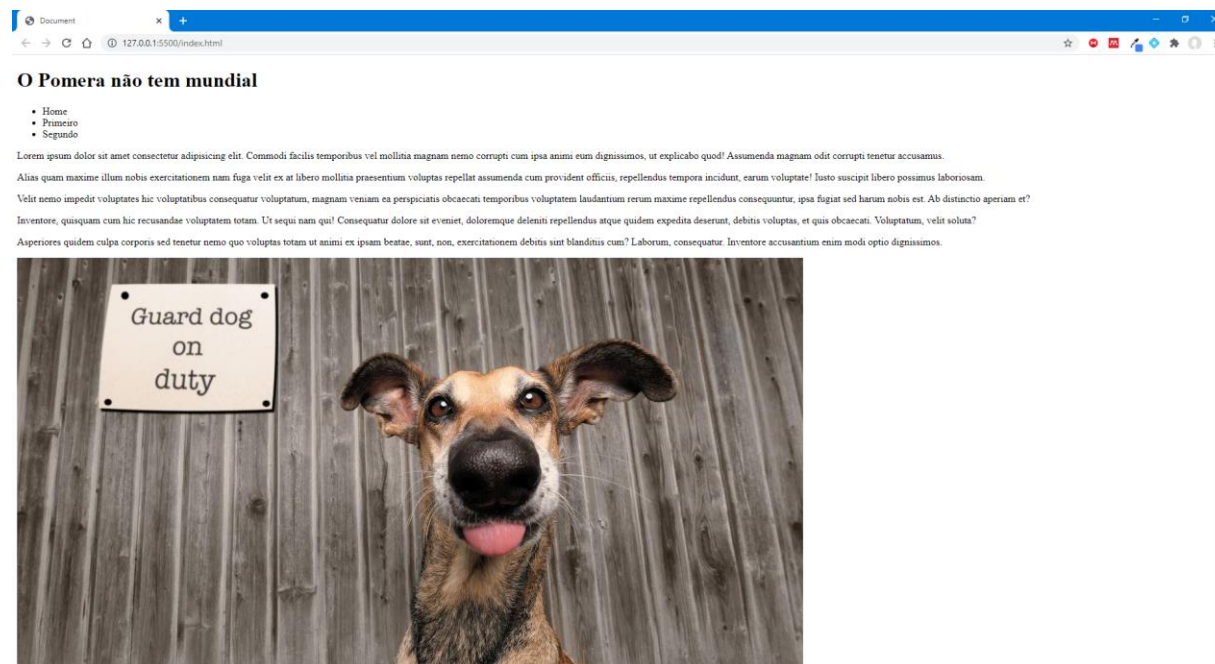


Vantagem: consome menos memória

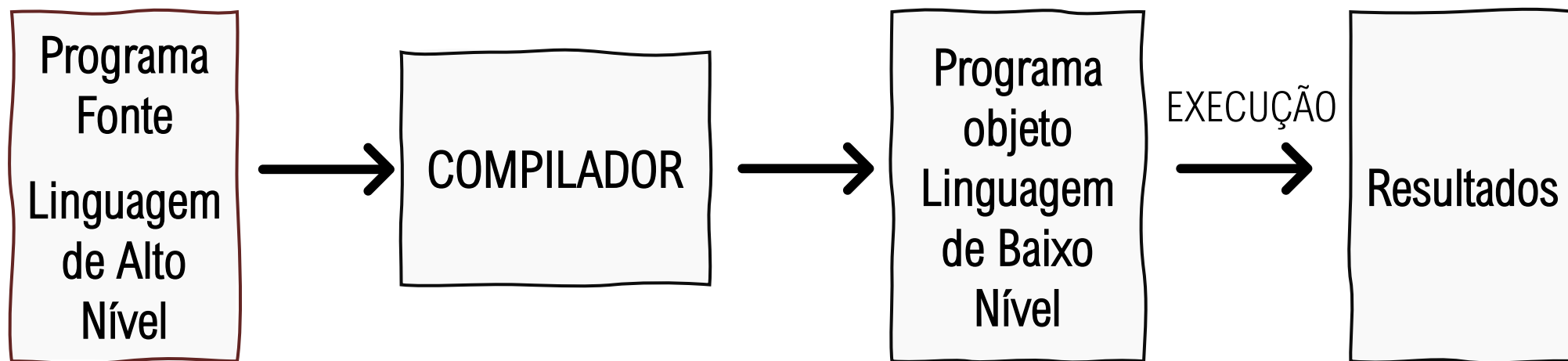
Desvantagem: execução mais lenta

- ✔ Traduz e faz a checagem da sintaxe e envia para execução, instrução por instrução.
- ✔ Precisa estar presente todas as vezes que vamos executar o programa e o processo acima é repetido.

Exemplo: Uma página HTML é interpretada pelo Navegador.



- ✔ Traduz o programa escrito em uma linguagem de programação para um programa equivalente escrito em linguagem de máquina (programa-objeto).



Vantagens:

- ✔ Velocidade de execução
- ✔ Oculta o código fonte

Desvantagem:

- ✔ A cada alteração no programa fonte é necessário gerar novamente o programa-objeto

- ✔ É considerada uma linguagem interpretada, pois os programas em Python são executados por um interpretador.
- ✔ Existem duas maneiras de usar o interpretador:
 - no modo de linha de comando
 - no modo de script: arquivo .py



Vamos criar um programa que apresente uma mensagem, faça a leitura de dados fornecidos pelo usuário e imprima na tela. O algoritmo em pseudocódigo ficará assim:

```
algoritmo exemplo1
  literal resposta
  inicio
    escreva("Sejam bem vindos ao mundo da programação")
    escreva("Digite o seu nome: ")
    leia(resposta)
    escreva("Você digitou: " + resposta)
  fim
```



Vamos criar um programa que apresente uma mensagem, faça a leitura de dados fornecidos pelo usuário e imprima na tela. O programa em Python ficará assim:

```
1 print("Sejam bem vindos ao mundo da programação")  
2 resposta = input("Digite o seu nome: ")  
3 print("Você digitou: ", resposta)
```







- ✓ Todos os exercícios devem ser resolvidos em Python.
- ✓ O código Python pode ser feito no IDLE, no Repl.it, ou na ferramenta que você ache mais adequada e deve ser salvo um arquivo por exercício com a extensão .py
- ✓ Após finalizar todos os exercícios da aula, coloque-os em uma pasta com o nome da aula, compacte a pasta e envie no Blackboard.



<https://youtu.be/BuxuUbgKwCg>



- 1- Faça um programa em Python que escreva a mensagem “Ciências da Computação - Unicsul” na tela.
- 2- Faça um programa em Python que solicite ao usuário a sua profissão e mostre na tela a informação digitada.
- 3- Faça um programa em Python que solicite ao usuário a sua idade e apresente a informação na tela.



- 4- Faça um programa em Python que solicite o último sobrenome do usuário e mostre na tela a mensagem: “Família” e na sequência o sobrenome digitado.
- 5- Faça um programa em Python que solicite o esporte favorito de uma pessoa e apresente-o na tela.



Esta aula foi elaborada com base no material produzido e cedido gentilmente pelos **Professores Alcides, Lédon, Amilton e Cristiane.**





That's all Folks!