



Programação de Computadores

- ✓ Conceitos iniciais

Motivação

- ✔ 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.



O que é um Programa de Computador?

- ✔ **Programa** é uma coleção de instruções que descrevem uma tarefa a ser realizada por um computador.
- ✔ Um programa é a formalização de um algoritmo em qualquer **linguagem de programação**, capaz de ser transformado em instruções que serão executadas por um computador, gerando os resultados esperados.

Fonte: Wikipédia http://pt.wikipedia.org/wiki/Programa_de_computador

- ✔ Sequência completa de instruções a serem executadas por computador.

Fonte: Miniaurêlio Eletrônico versão 5.12 (2004)

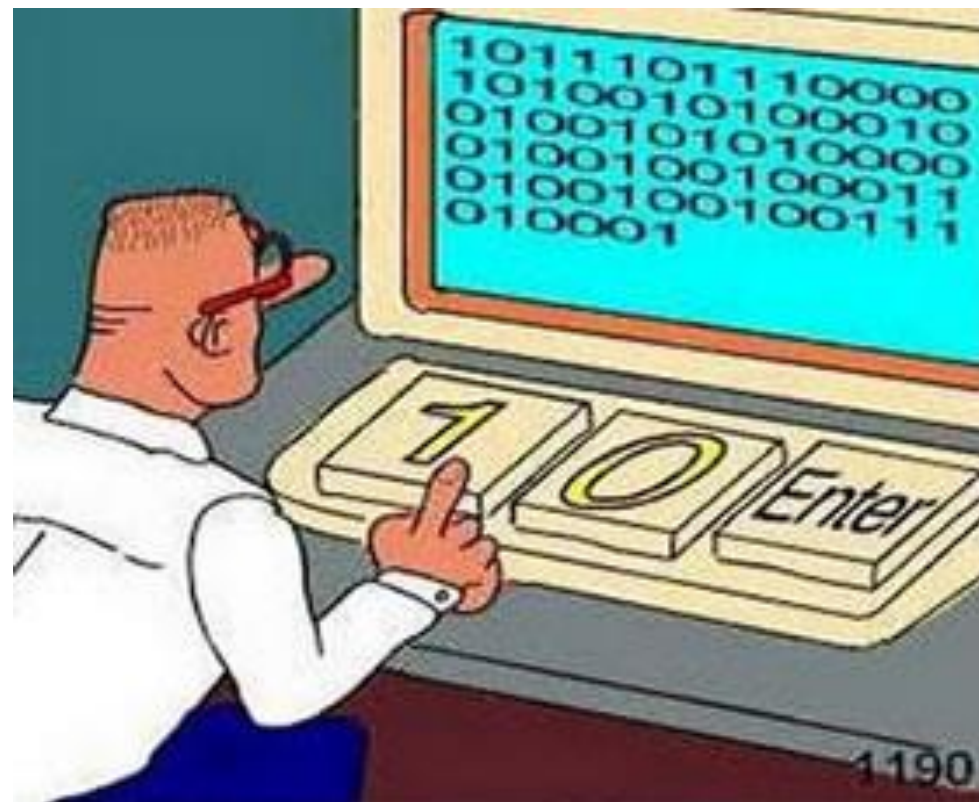


O que é um Programa de Computador?

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

```
01010101 01101101 00100000 01100101  
01111000 01100011 01100101 01101100  
01100101 01101110 01110100 01100101  
00100000 01110011 01100101 01101101  
01100101 01110011 01110100 01110010  
01100101 00100000 01110000 01100001  
01110010 01100001 00100000 01110100  
01101111 01100100 01101111 01110011  
00100001
```

Código de máquina (binário)



<https://www.invertexto.com/codigo-binario>

Como é construído?

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.



Algoritmo?

- ✓ 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.


Série de passos,
regras ou
procedimentos para
tentar solucionar um
problema



Algoritmo no dia-a-dia

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.

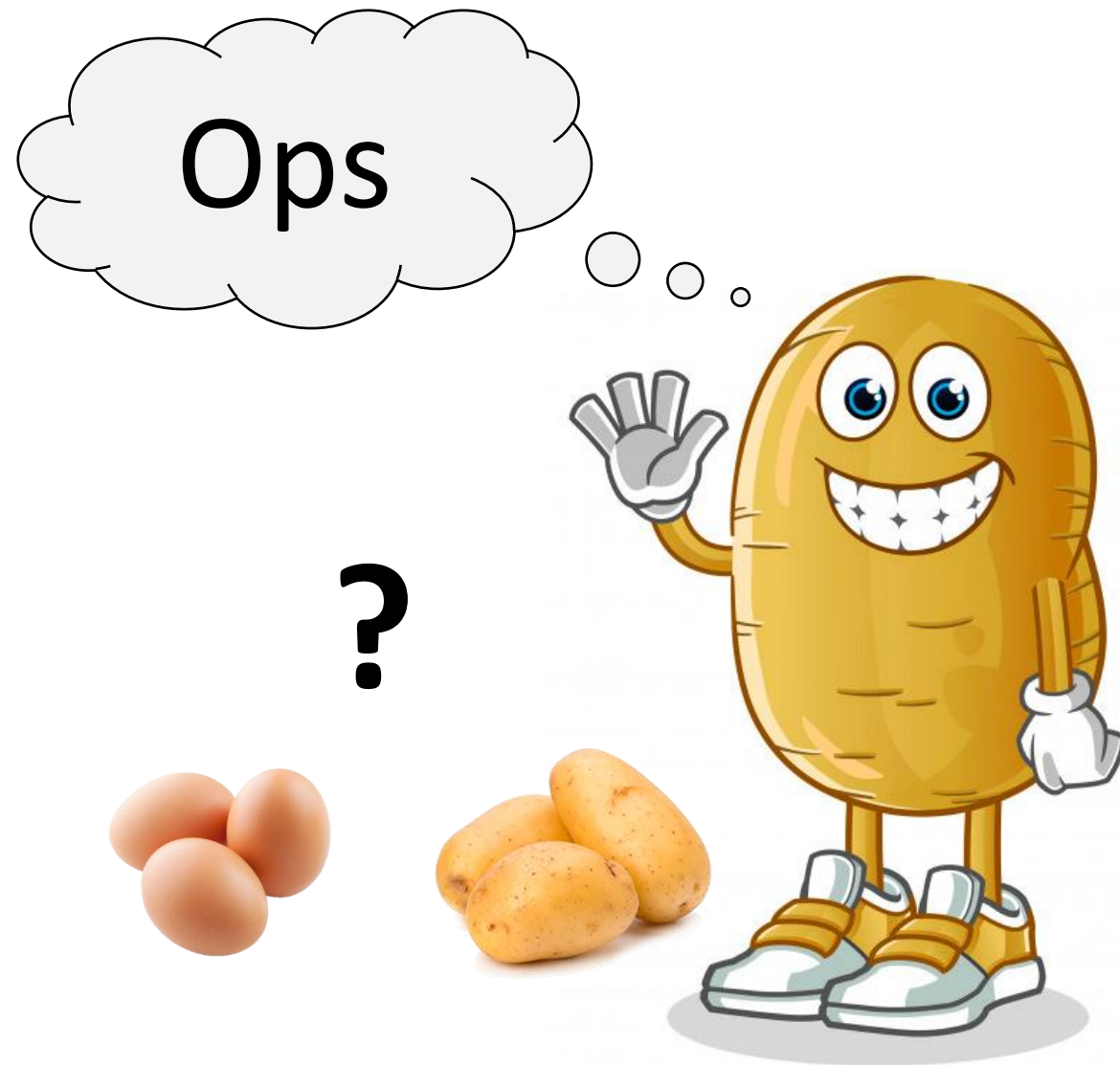
- 
- descer da escada
 - guardar a escada

muitas vezes
podemos
aprimorar um
algoritmo...



FORBELLONE, A. L. V.; EBERSPACHER, H. F. **Lógica de Programação: A Construção de Algoritmos e Estrutura de Dados**. 3. ed. São Paulo: Pearson Prentice Hall, 2008.

Algoritmo no dia-a-dia



Algoritmos – Como eles te controlam

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

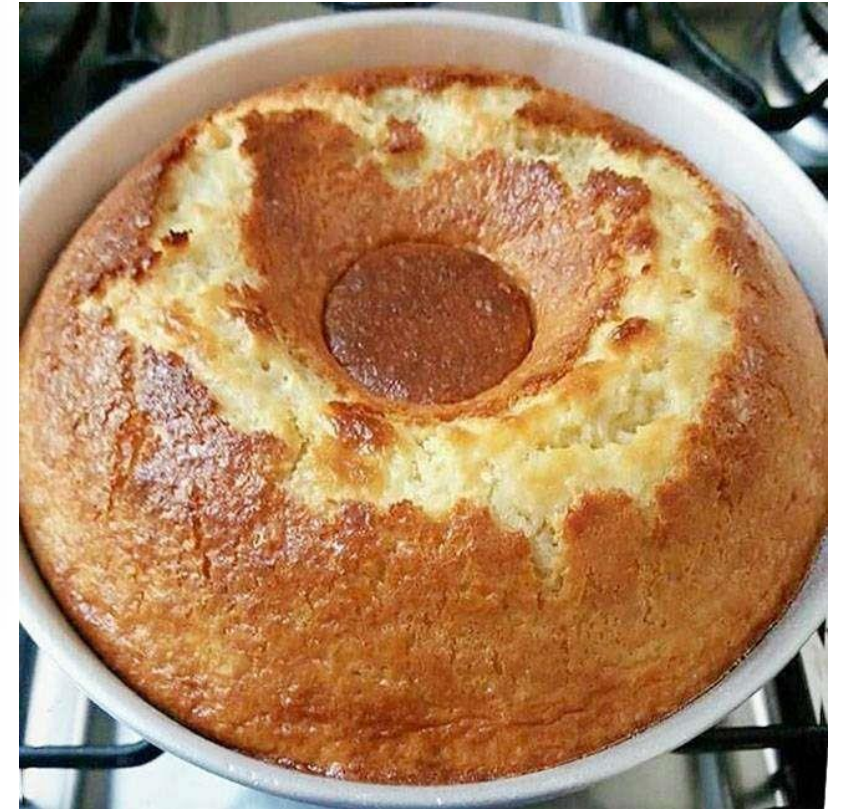


A estrutura de um algoritmo em português coloquial

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.

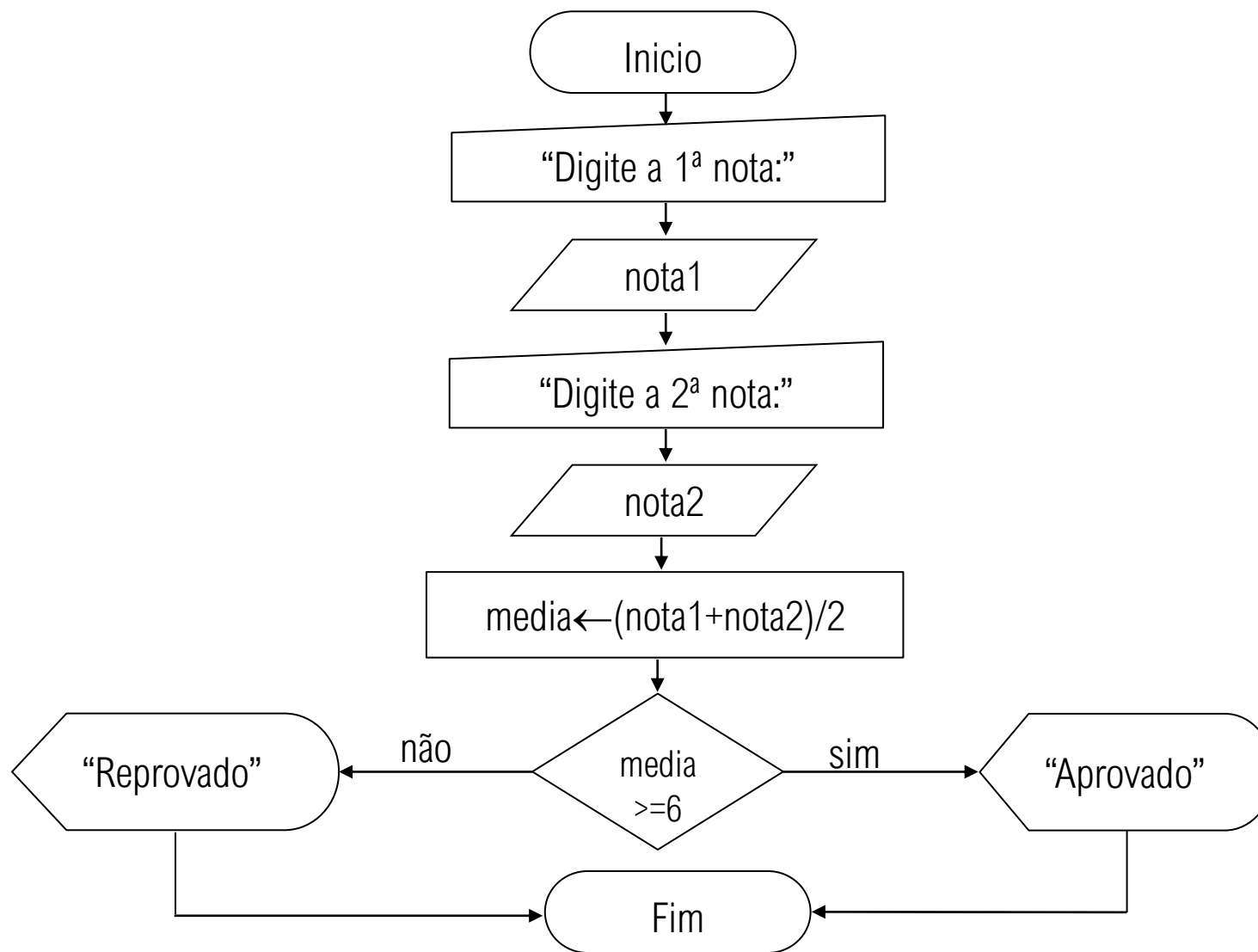


Algoritmo – Representação Pseudocódigo (Portugol)

```
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
```



Algoritmo – Representação Gráfica (Fluxograma)



Por que o algoritmo é importante ?

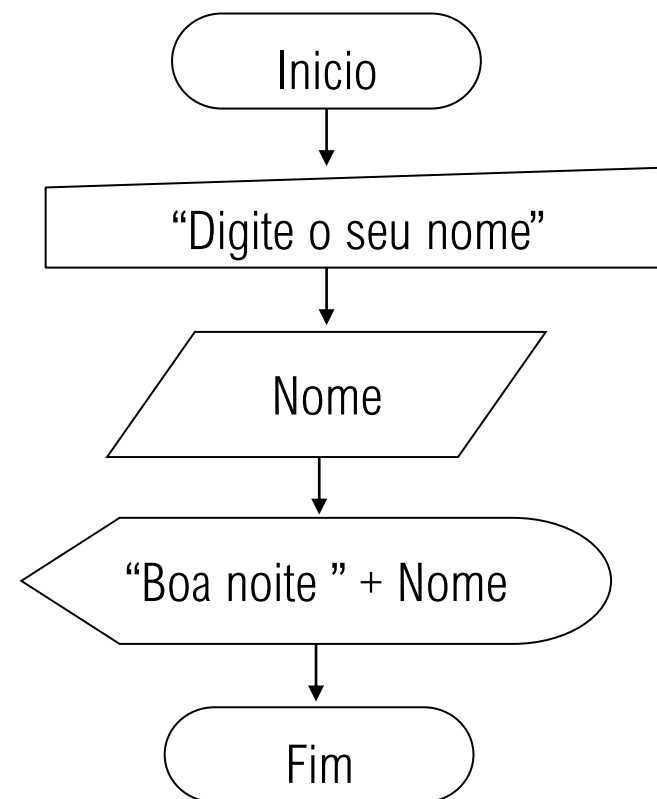
- ✔ Porque prepara uma lógica adequada, correta, para resolver um determinado problema.
- ✔ A partir dele será construído um programa, que estará escrito em alguma linguagem de programação para que possa ser executado em um computador.
- ✔ Pode ser programado em diferentes linguagens de programação.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



Exemplo 1

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
```



Linguagem Java

```
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 }
```



Linguagem C

```
#include<stdio.h>

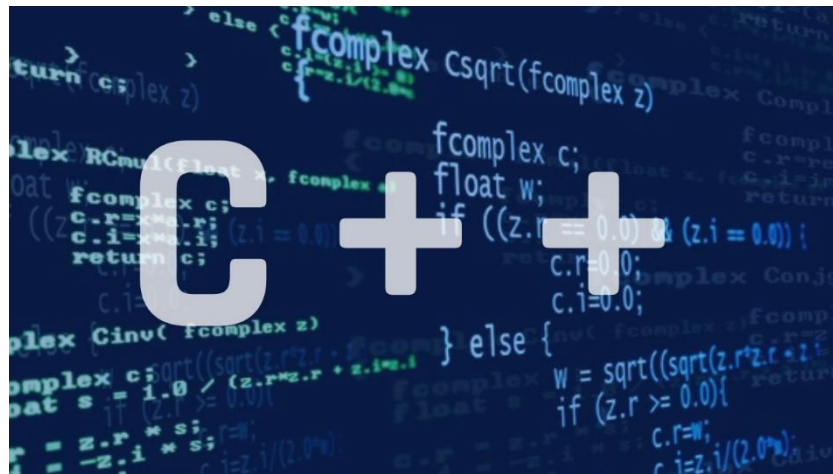
void main()
{
    char nome[30];
    printf("Digite o seu nome: ");
    gets(nome);
    printf("Boa Noite %s", nome);
}
```



Linguagem C++

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

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



Linguagem JavaScript

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

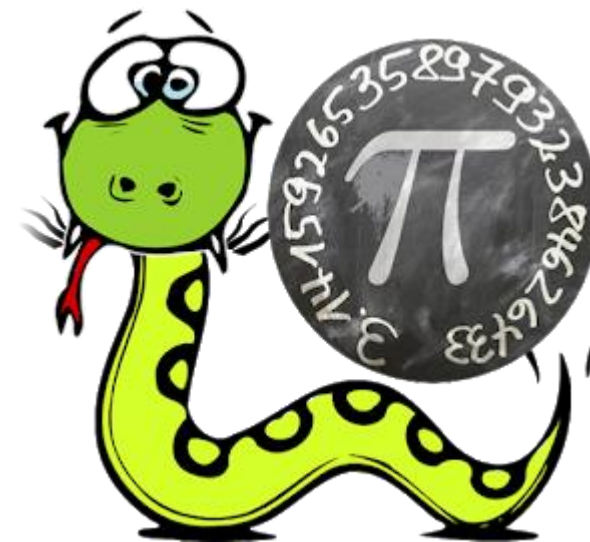
A yellow square with the letters 'JS' in a bold, black, sans-serif font.

Linguagem Python

```
1 nome = input("Digite seu nome: ")
2 print("Boa noite ", nome)
```



- ✓ A função **print()** exibe uma mensagem na tela do computador.
- ✓ A função **input()** solicita dados do usuário. Ela recebe um parâmetro, que é a mensagem a ser exibida, e retorna o valor digitado pelo usuário.



Como a máquina entende os códigos?

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.



Como a máquina entende os códigos?

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.

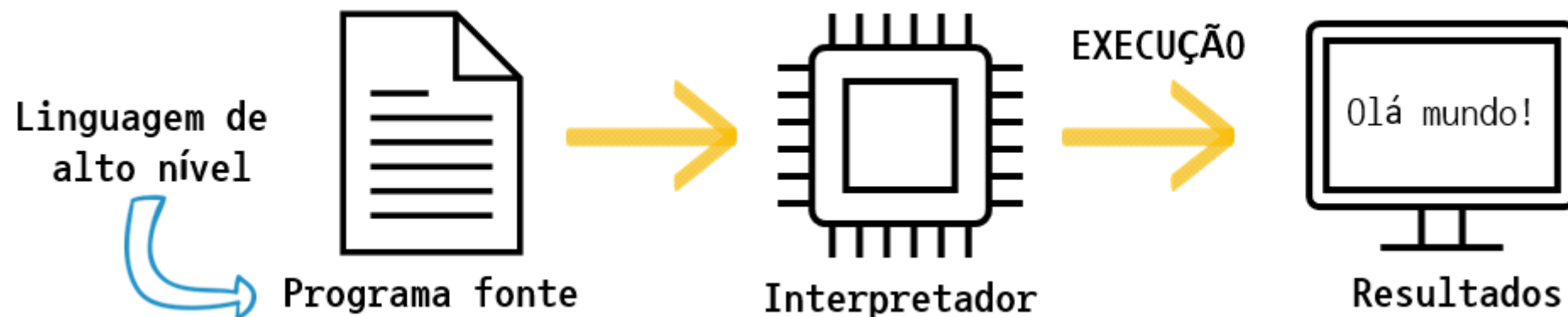
Para essa tarefa temos basicamente dois métodos:

- ✓ Compilador
- ✓ Interpretador



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.

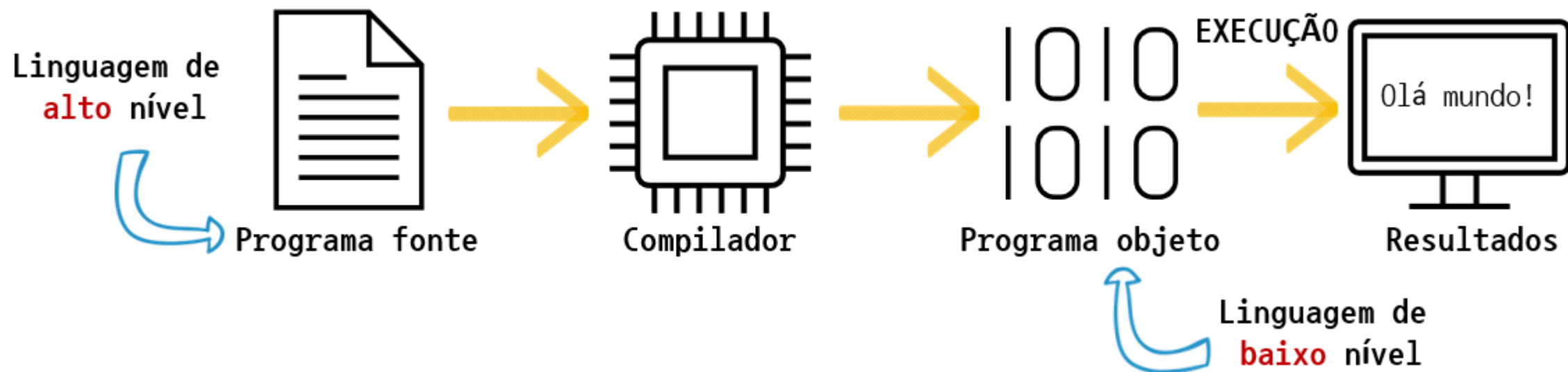


Vantagem: consome menos memória

Desvantagem: execução mais lenta

Compilador

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



Vantagens:

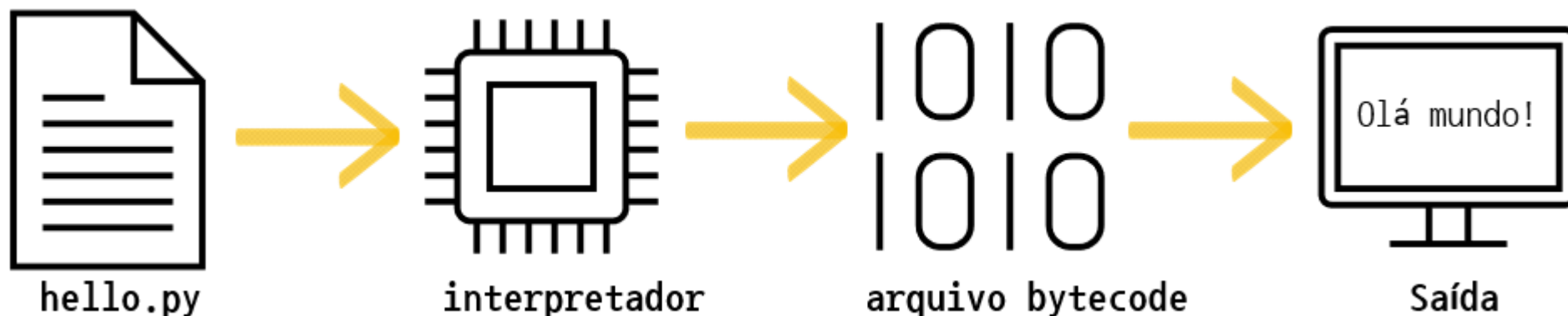
- ✓ Velocidade de execução
- ✓ Oculta o código fonte

Desvantagem:

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

Linguagem Python

Python: é uma linguagem de programação de propósito geral, de alto nível, interpretada, multiparadigma, de tipagem dinâmica e forte.



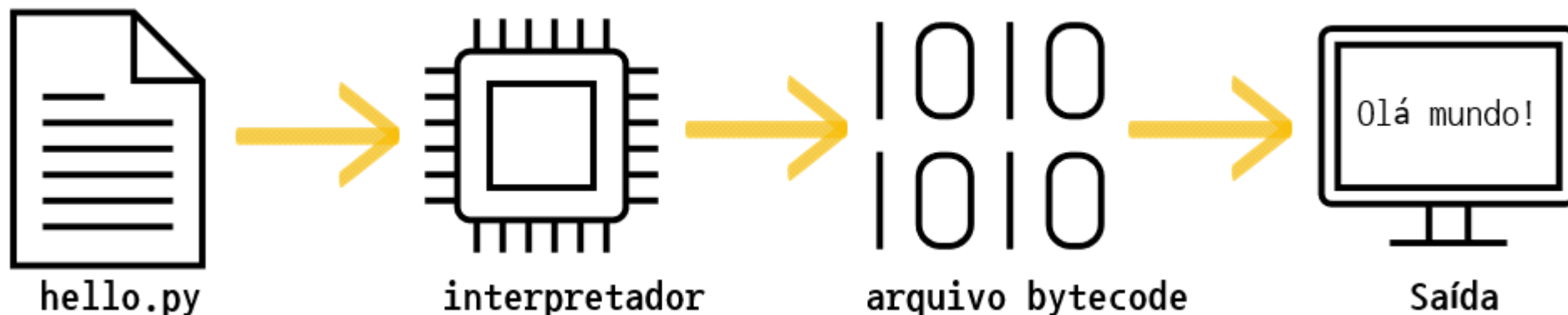
O processo para gerar a saída envolve os seguintes passos:

- Escrever um programa em Python
- Salvar o arquivo com a extensão `.py`
- Interpretar o código
- Gerar um arquivo de bytecode
- Imprimir a saída na tela



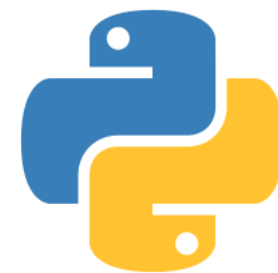
Linguagem Python

Python: é uma linguagem de programação de propósito geral, de alto nível, interpretada, multiparadigma, de tipagem dinâmica e forte.



Existem duas maneiras de usar o interpretador:

- no modo de linha de comando
- no modo de script: arquivo .py



Exemplo 2 - Python

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
```



Exemplo 2 - Python

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:

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



Alguma dúvida????



Exercícios de aplicação



Observações sobre exercícios

- ✓ 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>



Exercícios de aplicação

- 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.



Exercícios de aplicação

- 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.



Créditos

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





That's all Folks!