

**M01**

**[Em Revisão]**

# Fundamentos de Programação

Prof. Romário Nzenguele da Silva



**Licenciatura em Informática**

# Sobre o Documento

- ✓ Material de apoio **gratuito**, desenvolvido exclusivamente para os estudantes do 1º Ano da Licenciatura em Informática;
- ✓ O conteúdo, sempre que necessário, será atualizado para refletir as mudanças e melhorias no decorrer do curso. Cada atualização será devidamente **versionada**, permitindo o fácil acompanhamento das revisões;
- ✓ Este recurso é um **resumo das aulas**, essencial para apoiar no aprendizado da disciplina;
- ✓ **Sugestões de melhoria podem ser apresentadas pelo email que se encontra no rodapé.**

# Conteúdo

## 1. Introdução à Programação e Fundamentos

1.1. Conceito de Programação

1.2. Computadores e Programação

1.3. Linguagens de Programação: Definição e Classificação

1.4. Algoritmo vs Programa

### Objectivos

- ✓ Compreender o conceito de Programação
- ✓ Entender a relação entre Computadores e Programação
- ✓ Classificar Linguagens de Programação
- ✓ Diferenciar Algoritmo, Linguagem de Programação e Programa



# 1. Introdução à Programação e Fundamentos

Uma abordagem crucial para entender os conceitos fundamentais de programação

# 1.1 Conceito de Programação

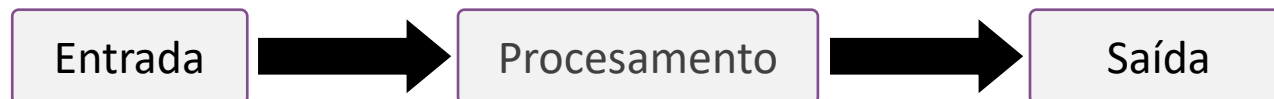
**Processo de comunicação com o computador por meio de instruções (código) para realizar tarefas específicas.**

- **Importância:** essencial para criar as tecnologias que usamos todos os dias, como sistemas operacionais, aplicativos de comunicação e entretenimento. Ela facilita tarefas e impulsiona inovações em áreas como inteligência artificial e IoT, sendo crucial para o avanço tecnológico e econômico.
- **Áreas de Aplicação:** desenvolvimento web/mobile/desktop, análise de dados, inteligência artificial, automação de processos ...

## 1.2 Computadores e Programação

- **Funcionamento do Computador:** Hardware + Software.

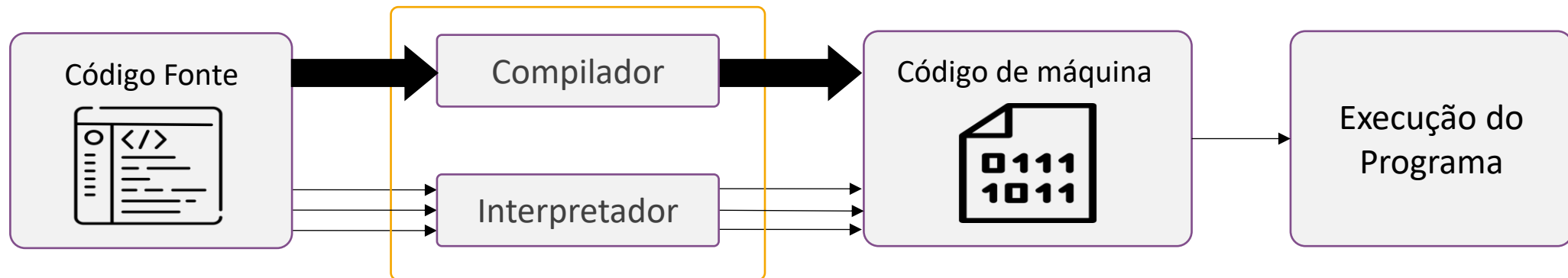
**Ex:** O hardware abrange a CPU, memória RAM, disco rígido, ..., enquanto o software refere-se ao sistema operacional (Windows, Linux) e outros programas, que gerenciam o uso do hardware e permitem a interação do usuário.



- **Interação:** interfaces gráficas e/ou linhas de comandos.

# 1.2 Computadores e Programação

- **Processo de execução de um programa:** conversão do **código fonte** em **código de máquina** -> execução do programa.
  - ✓ Compilação: o código é executado (compilado) na mesma instância; -> Processo realizado pelo **Compilador**.
  - ✓ Interpretação: o código é executado (interpretado) linha por linha. -> Processo realizado pelo **Interpretador**.



## Nota:

- ✓ **Código Fonte:** conjunto de instruções escritas através de uma linguagem de programação;
- ✓ **Código de máquina:** conversão do anterior em linguagem compreendida pelo computador -> Código binário.

## 1.3 Linguagens de Programação

- **Conceito:** regras + sintaxes -> programas

**Ex:** Python, Java, C/C++, JavaScript, Php, C#, Kotlin, Rust, Assembly, Go ...

- **Classificação:**

- Baixo nível: mais próximas do Hardware. Oferecem controlo directo sobre os recursos do sistema. Ex: Assembly ...
- Alto nível: mais próximas da linguagem do homem. Oferecem abstrações que facilitam a programação. Ex: Python ...



# 1.3 Linguagens de Programação

- **Linguagens Interpretadas vs Compiladas:**

- ✓ Compiladas: obedecem o processo de **compilação**, explicado anteriormente. Ex. C++, Kotlin, C#, ...
- ✓ Interpretadas: obedecem o processo de **interpretação**, explicado anteriormente. Ex. Php, Python, ...

- **Paradigmas de programação:** modelo (abordagem) de programação

**Ex:** imperativo (estruturado, procedural), declarativo (funcional, lógico), Orientado a Objectos, ...

**Nota:** teremos um espaço dedicado sobre os **paradigmas de programação**.

# 1.4 Algoritmo vs Programa

- **Algoritmo:** sequência de passos finitos -> resolução de um problema
- **Programa:** implementação do algoritmo numa linguagem de programação

Ex. Programa para somar dois números:

## Algoritmo

1. Receber o primeiro número.
2. Receber o segundo número.
3. Somar os dois números.
4. Exibir o resultado.



## Código fonte em “Python”

```
# Programa que soma dois números
num1 = int(input("Digite o primeiro número: "))
num2 = int(input("Digite o segundo número: "))
soma = num1 + num2
print(f"A soma é: {soma}")
```

**Nota:** o termo **programa** é comumente utilizado para referir-se ao **código fonte**. Por outro lado, para **usuários finais** refere-se ao **resultado da execução do código fonte**.

## Código fonte em “C”

```
#include <stdio.h>

int main() {
    int num1, num2, soma;

    // Solicitar ao usuário para inserir o primeiro número
    printf("Digite o primeiro número: ");
    scanf("%d", &num1);

    // Solicitar ao usuário para inserir o segundo número
    printf("Digite o segundo número: ");
    scanf("%d", &num2);

    // Calcular a soma
    soma = num1 + num2;

    // Exibir o resultado
    printf("A soma é: %d\n", soma);

    return 0;
}
```

## Código fonte em “JavaScript”

```
// Solicitar ao usuário para inserir o primeiro número
let num1 = parseInt(prompt("Digite o primeiro número: "));

// Solicitar ao usuário para inserir o segundo número
let num2 = parseInt(prompt("Digite o segundo número: "));

// Calcular a soma
let soma = num1 + num2;

// Exibir o resultado
alert(`A soma é: ${soma}`);
```



**Continua ... Sequência -> M02**