



**UNIVERSIDADE EDUARDO MONDLANE**  
**FACULDADE DE ENGENHARIA**  
**DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA**

# **COMPILADORES**

Introdução aos Compiladores

**Docentes:** Ruben Moisés Manhiça  
Cristiliano Maculuve

**Maputo, 2/28/2024**



# Conteúdo da Aula

1. Historia do surgimento dos compiladores;
2. Definição de compiladores;





# Um Pouco de História

- Uma linguagem de programação é um conjunto de ferramentas, regras de sintaxe e símbolos ou códigos que nos permitem escrever programas de computador.
- A primeira e mais primitiva linguagem de computador é a própria linguagem máquina (0's e 1's).
- Um programa era difícil, longo e principalmente caro de o construir.
- Era também difícil de ser entendido por outros programadores.
- Essa complexidade levou à necessidade de desenvolver novas técnicas e ferramentas.





# Um Pouco de História





# Um Pouco de História

- A resolução do problema passou pela criação de uma linguagem em que os códigos numéricos foram substituídos por **mnemónicos**.
- O nome dessa linguagem é **ASSEMBLY LANGUAGE**.
- Então será necessário um outro programa que leia o programa escrito nessa linguagem alternativa e o traduza para a linguagem nativa do computador!!!
- O processo de tradução da linguagem de montagem para a linguagem de máquina é realizada por um programa chamado **ASSEMBLER**.





# Um Pouco de História

Foram desenvolvidas diversas linguagens de programação:

- FORTRAN (1957)
- ALGOL (1958)
- COBOL (1959)
- PASCAL (1963)
- BASIC (1965)
- ADA (1968)
- DoD (1969)
- C (1982) e mais tarde o C++ (1986)
- Java;
- C#;
- Etc....

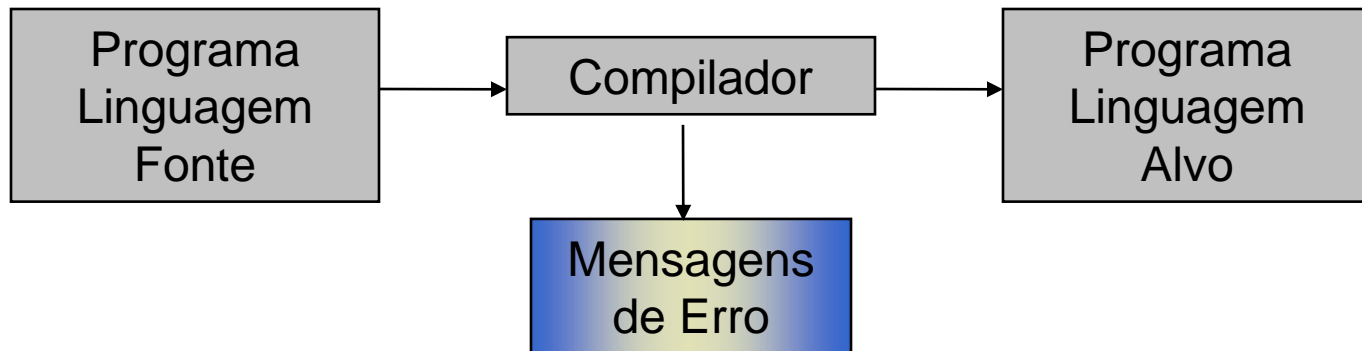
Estas novas linguagens foram afastando cada vez mais o programador do nível de máquina.





# O que é um Compilador?

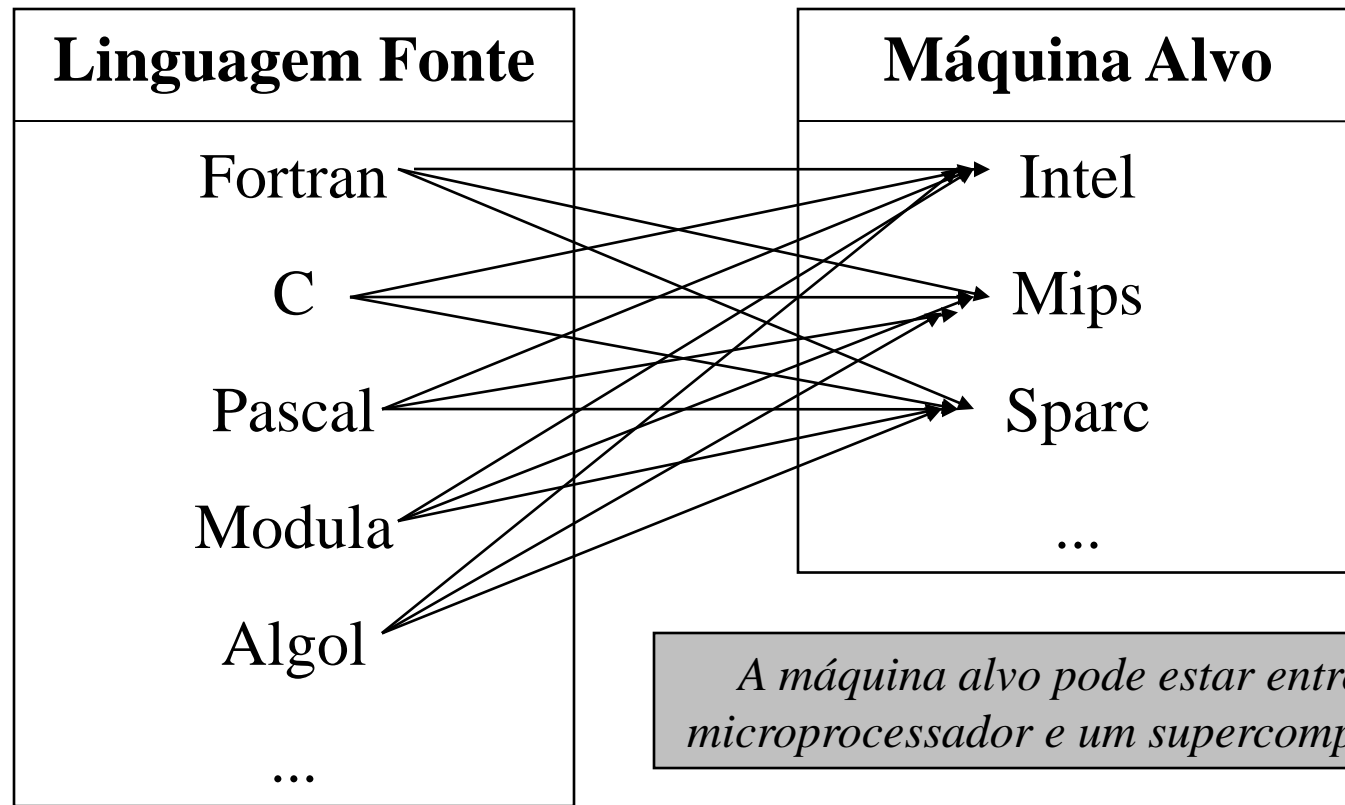
“Um compilador é um programa que lê um programa escrito em uma linguagem (linguagem fonte) e a **traduz em um** programa equivalente em outra linguagem (linguagem alvo).” Aho, Sethi, Ullman





# O que é um Compilador?

## VARIEDADE DE COMPILADORES







# O que é um Compilador?

Nesse processo de compilação, há duas tarefas básicas a serem executadas por um compilador:

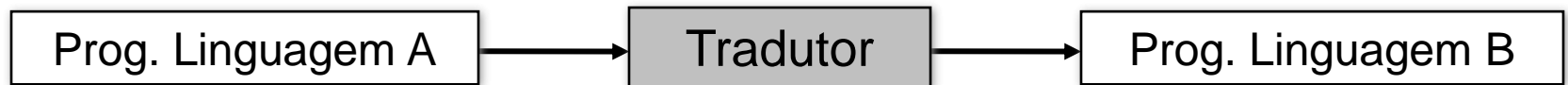
- **Análise**, *em que o texto de entrada (na linguagem fonte) é examinado, verificado e compreendido*
  - Análise léxica, sintática e semântica
- **Síntese**, *ou geração de código, em que o texto de saída (na linguagem objeto) é gerado, de forma a corresponder ao texto de entrada*



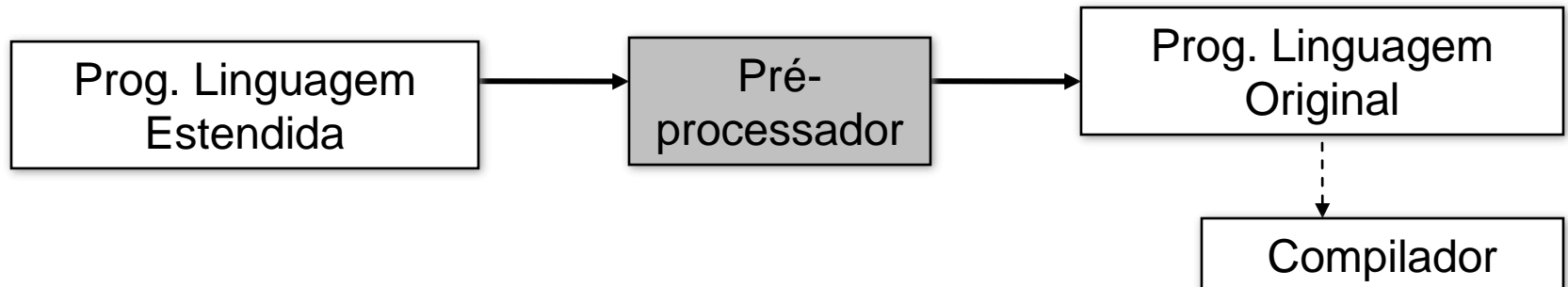


# Conceitos acerca da compilação

**Tradutor**: programa que transforma um programa fonte escrito numa linguagem em um programa equivalente escrito em uma linguagem diferente;



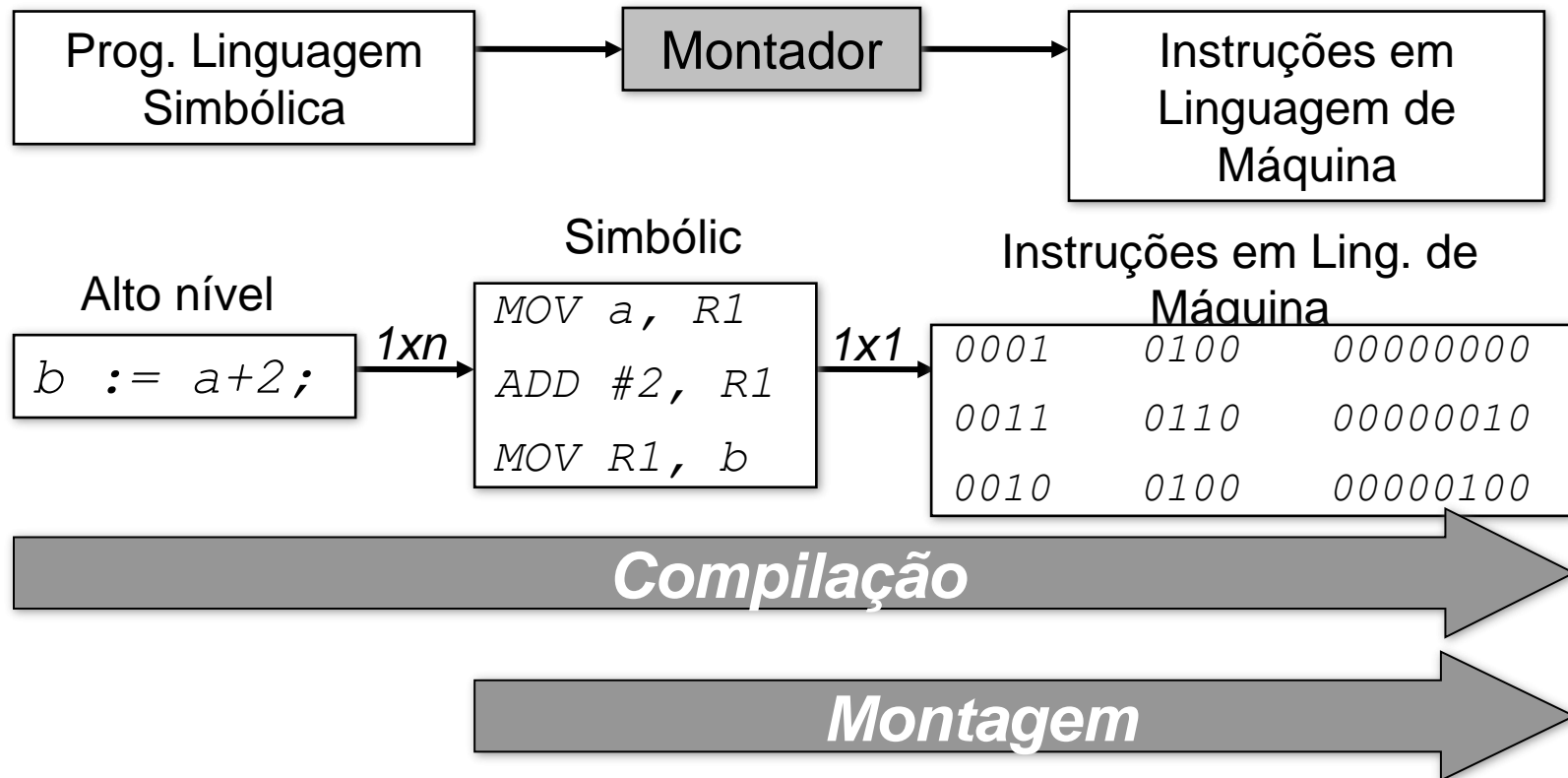
**Pré-processador**: programa que transforma um programa escrito em uma linguagem estendida em um programa equivalente escrito em linguagem original;





# Conceitos acerca da compilação

**Montador**: tradutor que transforma um programa escrito em uma linguagem simbólica (de baixo nível) em instruções equivalentes em linguagem de máquina;





# Conceitos acerca da compilação

**Interpretador**: tradutor que funciona em tempo de execução. Estes programas traduzem programas codificados em linguagem de alto nível para um código intermediário e o coloca em execução.

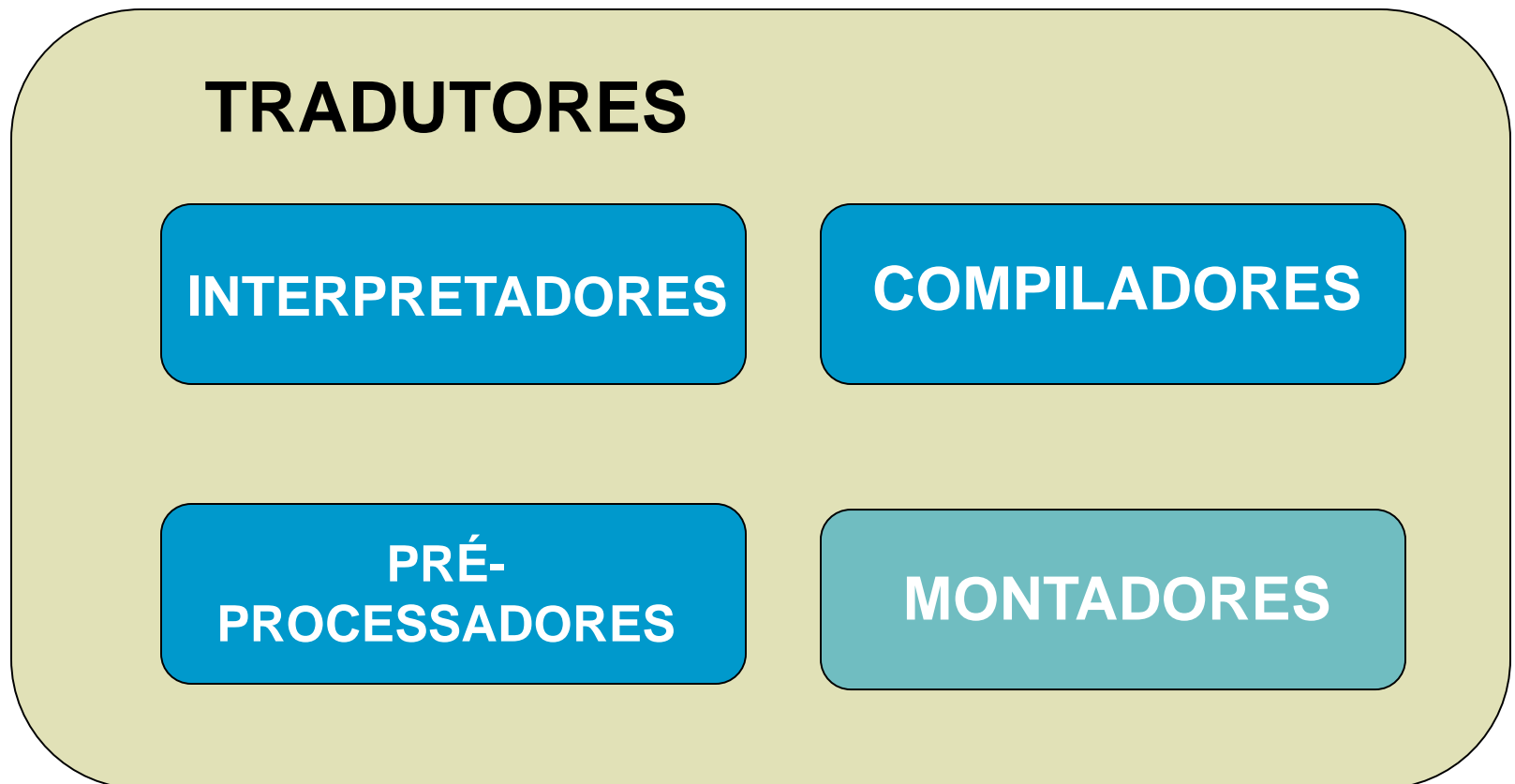
- Exemplos:
  - o *run* do Pascal ou C;
  - Java: sistemas distribuídos, diferentes equipamentos; (bytecode)
    - » Cerca de 10 vezes mais lento que um código compilado C++;
    - » Pode ser interpretado em plataformas diferentes;
  - Maior parte das linguagens para Browsers de Internet;





# Conceitos acerca da compilação

Hierarquia:





# TPC

- Estudar as diferenças entre Interpretadores e Compiladores;
- Ver quais são as principais etapas da compilação;
- Discutir em conjunto com os outros colegas sobre a linguagem a ser usada no projecto da disciplina;
- Trazer para a aula pratica exemplos de compiladores:
  - Analisar a linguagem em que foram implementados;



**FIM!!!**

Duvidas e Questões?

