Compiladores

Análise Léxica

Bruno Lopes

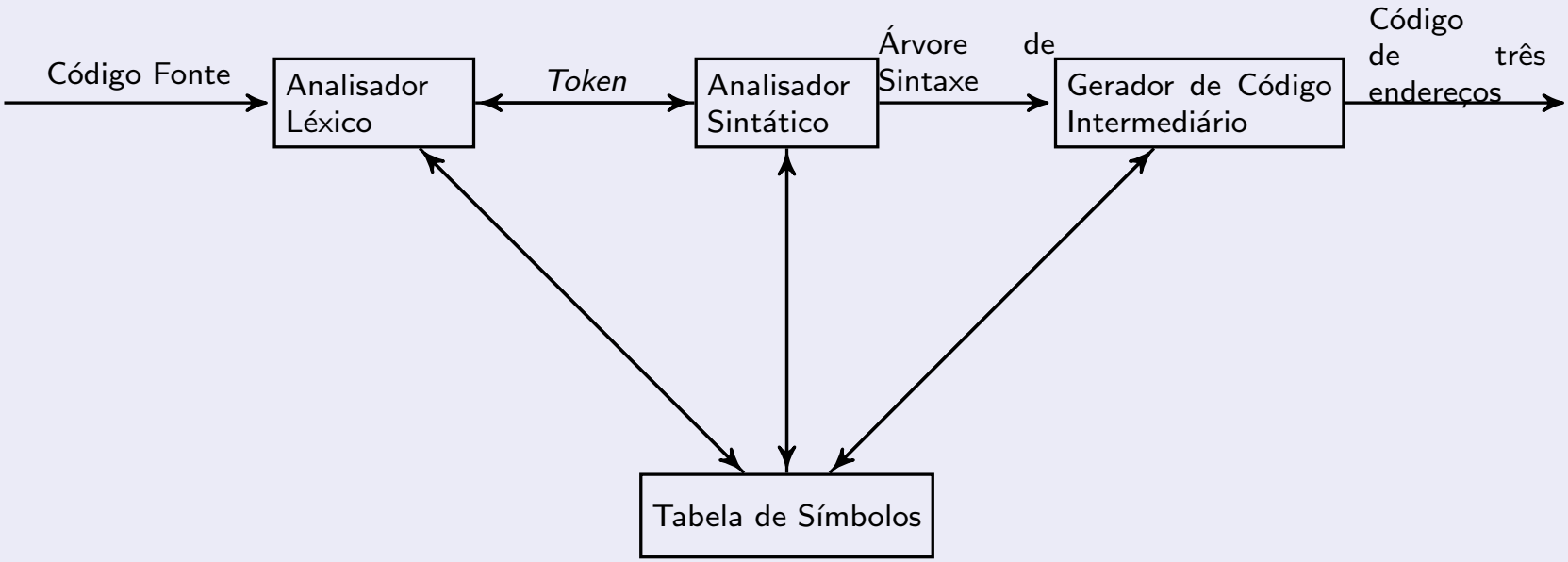
Instituto de Computação
UFF

Front-end

- Lida com a linguagem de entrada
- Teste de pertinência: código fonte $\in$ linguagem fonte?
- Programa está bem formado?
  - Sintaticamente?
  - Semanticamente?
- Cria um código intermediário

Instituto de Computação

Front-end



Instituto de Computação

Front-end

Implementação...

Scanning

- Expressões Regulares
- Autômatos Finitos Determinísticos

Parsing

- Gramáticas Livres de Contexto
- Autômatos de Pilha

Instituto de Computação

Front-end

Implementação...

Scanning

- Expressões Regulares
- Autômatos Finitos Determinísticos

Parsing

- Gramáticas Livres de Contexto
- Autômatos de Pilha

Instituto de Computação

Scanner

- Ler caracteres do código fonte
- Organizá-los em unidades lógicas (*tokes*)

```
if x == y then
  z = 1;
else
  z = 2;
```

Instituto de Computação

Scanner

- Ler caracteres do código fonte
- Organizá-los em unidades lógicas (*tokes*)

```
if x == y then
  z = 1;
else
  z = 2;
```

Instituto de Computação

Scanner

Quantos *tokens*?

```
if x == y then
  z = 1;
else
  z = 2;
```

Instituto de Computação

Scanner

Como definir e reconhecer tokens?

- como agrupar caracteres para formar as palavras da linguagem?
- como determinar se cada palavra é válida na linguagem fonte?

Instituto de Computação

Scanner

Funcionamento

- ler um fluxo de caracteres de entrada
- produzir um fluxo de saída que contém tokens, cada um rotulado com sua categoria sintática

Instituto de Computação

Categorias sintáticas

Palavras reservadas

IF, THEN, ELSE,...

Símbolos especiais

PLUS, MINUS, EQUAL,...

Cadeias de caracteres

ID, NUM, LITERAL,...

Instituto de Computação

Categorias sintáticas

Palavras reservadas

IF, THEN, ELSE,...

Símbolos especiais

PLUS, MINUS, EQUAL,...

Cadeias de caracteres

ID, NUM, LITERAL,...

Instituto de Computação

Categorias sintáticas

Palavras reservadas

IF, THEN, ELSE,...

Símbolos especiais

PLUS, MINUS, EQUAL,...

Cadeias de caracteres

ID, NUM, LITERAL,...

Instituto de Computação

Analisador Léxico

⟨tipo-token, token⟩

a = 10 ≡ ⟨ID, "a"⟩,⟨=, "="⟩, ⟨NUM, "10"⟩

Instituto de Computação

Analisador Léxico

⟨tipo-token, token⟩

a = 10 ≡ ⟨ID, “a”⟩,⟨=, “=”⟩, ⟨NUM, “10”⟩

Instituto de Computação

Analisador Léxico

Quantos tokens de cada tipo?

```
x = 0
while (x < 10 ){
  x++;
}
```

Instituto de Computação

Scanner

Microsintaxe

- Regras descrevendo a estrutura léxica da linguagem de entrada
- Geralmente são simples: caracteres em branco e marcas de pontuação terminam uma palavra, identificadores começam com letras,...

Instituto de Computação

Expressões Regulares

- Representam padrões de cadeias de caracteres
- Definida pelo conjunto de cadeias de caracteres com as quais ela casa

Instituto de Computação

Overview

1. Escrever expressão regular para representar a linguagem
2. Construir NFA para reconhecer a linguagem, a partir das ERs
3. Transformar NFA em DFA
4. Minimizar DFA
5. Construir o scanner a partir do DFA

Instituto de Computação

Expressões Regulares

- Representam padrões de cadeias de caracteres
- Definida pelo conjunto de cadeias de caracteres com as quais ela casa

Instituto de Computação