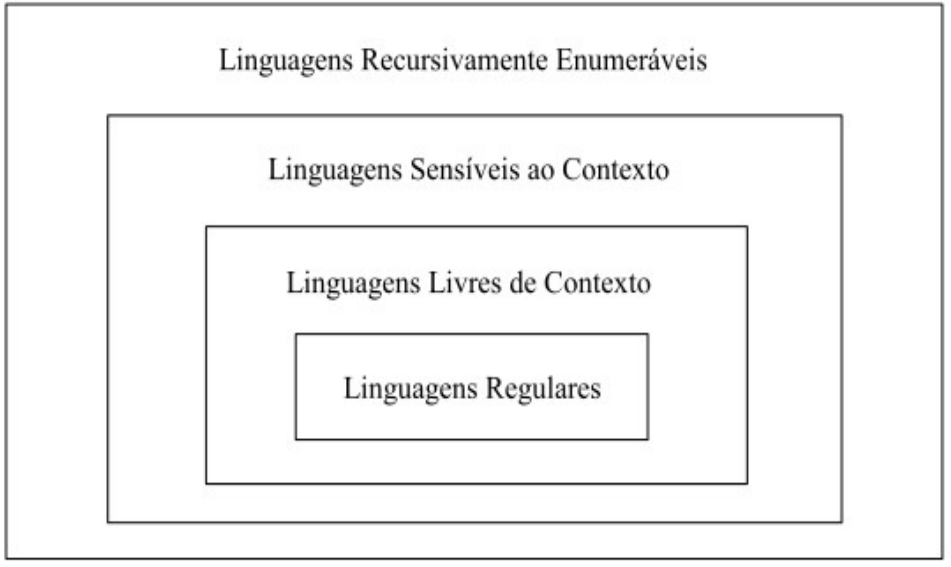
Linguagens Livres de Contexto

Introdução

- Hierarquia de Chomsky



Linguagens Livres de Contexto

- Formalização sintática das linguagens de programação de alto nível
- Representação de construções aninhadas
  - na construção de expressões aritméticas
  - na estruturação do fluxo de controle
  - na estruturação do programa

Definição

- Quádrupla (V, Σ, P, S) com os seguintes componentes
  - V: conjunto (finito e não-vazio) dos símbolos terminais e não-terminais
  - Σ: conjunto (finito e não-vazio) dos símbolos terminais; corresponde ao alfabeto da linguagem definida pela gramática
  - P: conjunto (finito e não-vazio) das regras de produção, todas no formato $\alpha \rightarrow \beta$, com $\alpha \in (V - \Sigma)$ e $\beta \in V^*$;
  - S: raiz da gramática, $S \in (V - \Sigma)$

Árvores de Derivação

- Melhor visualização da estrutura das sentenças da linguagem
  - facilitando a análise das mesmas
- Facilita a representação interna
  - nos compiladores e interpretadores

Árvores de Derivação

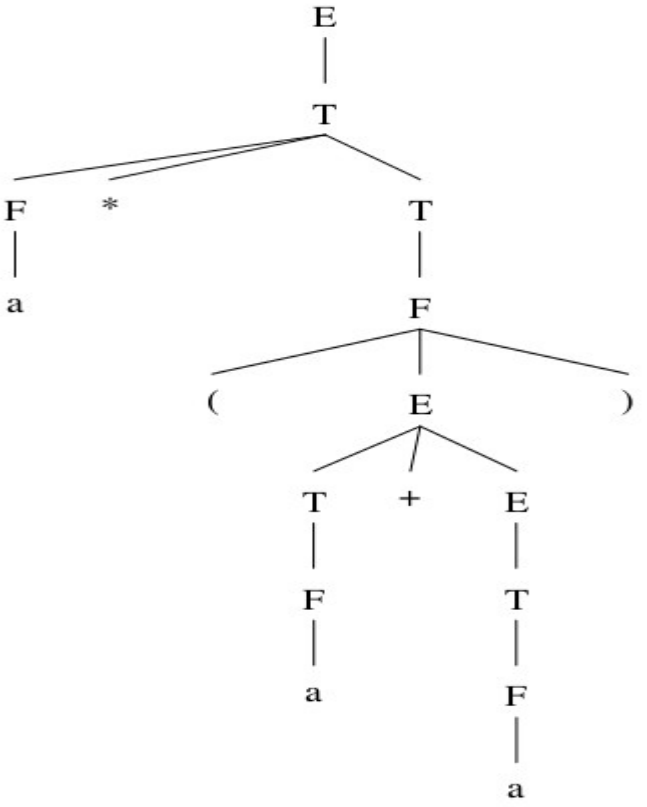
- Exemplo:
  - Sentença: a * (a + a)

$$
\begin{aligned}
\{E &\rightarrow T + E, \\
E &\rightarrow T, \\
T &\rightarrow F * T, \\
T &\rightarrow F, \\
F &\rightarrow (E), \\
F &\rightarrow a\}
\end{aligned}
$$

Árvores de Derivação

- Exemplo:
  - Sentença: a * (a + a)

$$
\begin{aligned}
\{E &\rightarrow T + E,\\
E &\rightarrow T,\\
T &\rightarrow F * T,\\
T &\rightarrow F,\\
F &\rightarrow (E),\\
F &\rightarrow a\}
\end{aligned}
$$



Árvores de Derivação

- Árvores de derivação não contêm informação sobre a sequência em que foram aplicadas as produções
  - elas informam apenas quais foram as produções aplicadas, mas não em que ordem

Gramática não-ambígua

- Se para toda e qualquer cadeia pertencente à linguagem, existir uma única sequência de **derivações mais à esquerda** e uma única sequência de **derivações mais à direita** que a geram

Exemplo 1

- Linguagem das expressões aritméticas sobre {a, +, * , (,)} com apenas um símbolo não-terminal E

$$\begin{aligned}\{E &\rightarrow E+E,\\ E &\rightarrow E*E,\\ E &\rightarrow a,\\ E &\rightarrow (E)\}\end{aligned}$$

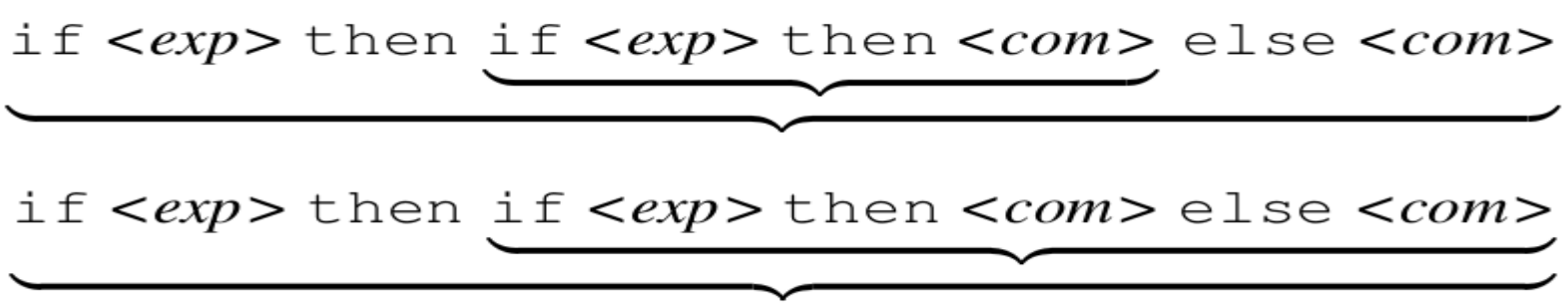
- Cadeia: a + a * a

Exemplo 1

- Aplicando-se inicialmente: $E \rightarrow E + E$
  - $E \Rightarrow E + E \Rightarrow a + E \Rightarrow a + E * E \Rightarrow a + a * E \Rightarrow a + a * a$
- Aplicando-se inicialmente: $E \rightarrow E * E$
  - $E \Rightarrow E * E \Rightarrow E + E * E \Rightarrow a + E * E \Rightarrow a + a * E \Rightarrow a + a * a$

Exemplo 2

- Ambiguidade



Linguagens Sensíveis ao Contexto

Gramática

- Gramática sensível ao contexto $G = (V, \Sigma, P, S)$
- Conjunto P obedecem ao formato $\alpha \rightarrow \beta$, onde:
  - $\alpha \in V^* N V^*$
  - $\beta \in V^*$
  - $| \beta | >= | \alpha |$

Definição

- Definição é estendida para qualquer linguagem L que contenha a cadeia vazia, desde que L − { ε } possa ser gerada por uma gramática sensível ao contexto;

Exemplo

- {Programa → Declaracoes Comandos,
- Declaracoes → Declaracoes Declaracao | ε ,
- Declaracao → “%”Identificador,
- Comandos → Comandos Comando | ε ,
- Comando → “#”Identificador“ = ”Expressao,
- Expressao → Expressao“ + ”Expressao
  | Expressao“ * ”Expressao
  | Identificador,
- Identificador → “a” | “b” | “c”}

Exemplo

- Exemplo que pertence a linguagem:

```
%a
%b
#a = a + b
#b = b * b
```

- Exemplo de sentença não pertencente a esta linguagem:

```
%a
%c
#a = a + b
#b = b * b
```

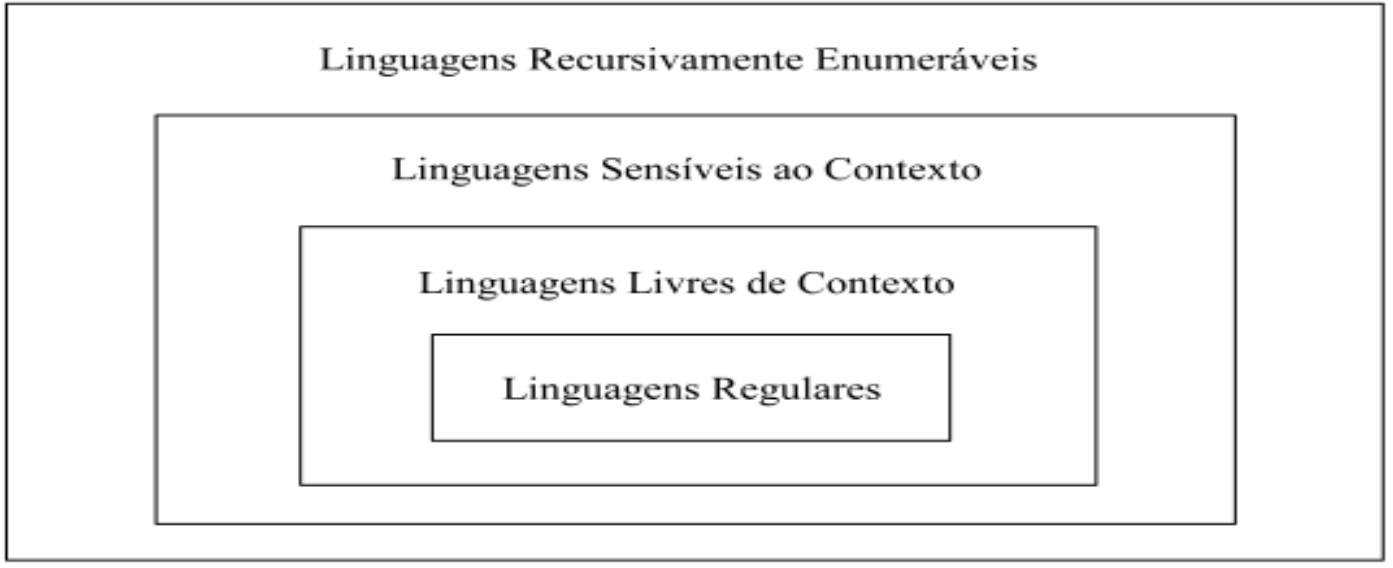
Formalização

- Possível, mas trabalhosa;
- Produz especificações longas, complexas e com baixa legibilidade;
- Difícil utilização prática;
- Por isso, adota-se a representação “livre de contexto” na formalização gramatical, deixando para processamento posterior a verificação das dependências de contexto que a linguagem porventura exiba.

Conclusão

Hierarquia de Chomsky

- Linguagens



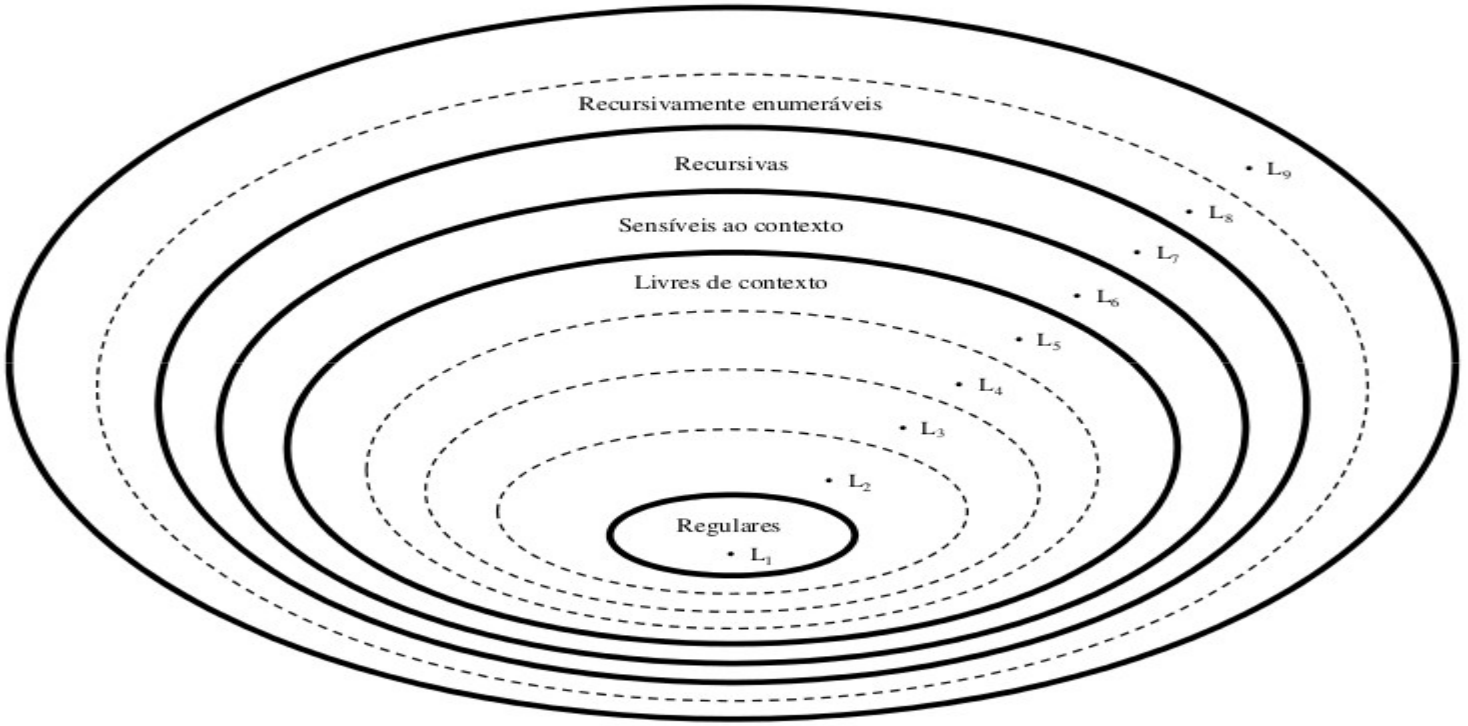
Hierarquia de Chomsky

- Reconhecedores:

Tipo	Classe de linguagens	Modelo de gramática	Modelo de reconhecedor
0	Recursivamente enumeráveis	Irrestrita	Máquina de Turing
1	Sensíveis ao contexto	Sensível ao contexto	Máquina de Turing com fita limitada
2	Livres de contexto	Livre de contexto	Autômato de pilha
3	Regulares	Linear (direita ou esquerda)	Autômato finito

Hierarquia de Chomsky

- Hierarquia de Inclusão



Hierarquia de Chomsky

Tipo (Hierarquia de Chomsky)	Classe de linguagens	Gramática	Reconhecedor	Reconhecedor determinístico ≡ não-determinístico?	Estruturas sintáticas típicas da classe de linguagens
3	Regular	Regular	Autômato finito	Sim	Repetição, união e concatenação de termos
2	Livre de contexto determinística descendente	LL(k)	Autômato de pilha determinístico	N.A.	Aninhamento de construções sintáticas
	Livre de contexto determinística ascendente	LR (k)	Autômato de pilha determinístico	N.A.	?
	Livre de contexto não-ambígua	Livre de contexto não-ambígua	Autômato de pilha	Não	?
	Livre de contexto	Livre de Contexto	Autômato de pilha	Não	?
1	Sensível ao contexto	Sensível ao contexto	Máquina de Turing com fita limitada	?	Dependência entre termos
0	Recursiva	?	Máquina de Turing que sempre pára	Sim	?
	Recursiva-mente enumerável	Irrestrita	Máquina de Turing	Sim	?
N.A	Não-gramaticais	N.A.	?	N.A.	?