Gramáticas Regulares

Formalmente, as gramáticas regulares são caracterizadas como quádruplas ordenadas

$$G = (V_n, V_t, P, S)$$

onde:

Vn representa o vocabulário não terminal da gramática. Este vocabulário corresponde ao conjunto de todos os símbolos dos quais a gramática se vale para definir as leis de formação das sentenças da linguagem.

Vt é o vocabulário terminal, contendo os símbolos que constituem as sentenças da linguagem. Dá-se o nome de terminais aos elementos de Vt.

P representa o conjunto de todas as leis de formação utilizadas pela gramática para definir a linguagem (regular).

Para tanto, cada construção parcial, representada por um não-terminal, é definida como um conjunto de regras de formação relativas à definicão do não-terminal a ela referente. A cada uma destas regras de formação que compõem o conjunto P dá-se o nome de produção da gramática.

Assumimos $Vn \cap Vt = \emptyset$. Convencionamos que $Vn \cup Vt = V$
Cada produção P tem a forma:

A -> aB ou A -> a (linear à direita)
OU
A -> Ba ou A -> a (linear à esquerda)
onde $A, B \in Vn$ e $a \in Vt$

S ϵ Vn denota a principal categoria gramatica de G; é dito o símbolo inicial ou o axioma da gramática. Indica onde se inicia o processo de geração de sentenças.

Ex.1: G = ({S, A, B}, {a, b}, P, S)
P: {S -> aA
A -> aA | bB
B -> bB | b}

Notação/Convenções

- Letras do alfabeto latino maiúsculas {A,B,..Z}: variáveis
- Letras do começo do alfabeto latino minúsculas {a,b,c,...}: terminais
- Letras do fim do alfabeto latino minúsculas {t,u,v,x,z}: cadeias de terminais
- Letras gregas minúsculas {α,β,γ,**δ,ε,** ..., ω}: cadeias de terminais e não terminais

alfa, beta, gama, delta, épsilon, zeta, eta, teta, iota, kapa, lâmbda, mi, ni, xi, ômicron, pi, rô, sigma, tau, úpsilon, fi, qui, psi, ômega

Definida uma gramática G, qual é a linguagem gerada por ela?

Precisaremos das relações =>G (deriva diretamente) e =>G* (deriva) definidas entre as cadeias de V*

Def1. Se $\alpha \rightarrow \beta$ é uma produção de P, dizemos que $\alpha \Rightarrow_G \beta$ (α deriva diretamente β na gramática G).

No Ex.1.: $S \Rightarrow_G aA \Rightarrow_G aaA$ ou
$S \Rightarrow_G aA \Rightarrow_G abB \Rightarrow_G abb$

Def2. Suponha que $\alpha_1\ \alpha_2\ \alpha_3\ \ldots\ \alpha_m$ são cadeias de V^* e $\alpha_1 =>_G \alpha_2$, $\alpha_3 =>_G \alpha_4$, ..., $\alpha_{m-1} =>_G \alpha_m$. Então dizemos que $\alpha_1 =>_G^* \alpha_m$ (deriva). Aplicamos algum número de produções de P. Por convenção $\alpha =>_G^* \alpha$ para a cadeia α.

No Ex.1.: $S =>_G^* abb$

$S =>_G^* abB$

$aaA =>_G^* aabbb$

$abb =>_G^* abb$

Def3. Forma sentencial: uma cadeia α composta de terminais e não terminais se $S \Rightarrow^* \alpha$

No Ex.1: S, aA, aaA, aabB, aabbb são formas sentenciais.

Uma forma sentencial, α, é uma sentença de G se $S \Rightarrow^* \alpha$ e $\alpha \in Vt^*$ (i.e., composta apenas de terminais). Ou seja, as cadeias geradas pela gramática são as sentenças de G.

Def4. A Linguagem L gerada por uma gramática G é definida como o conjunto de sentenças (cadeias de terminais) geradas por G. Ou seja,

$L(G) = \{x \mid x \in Vt^{*} \text{ e } S \Rightarrow_G^{*} x\}$ ou $\{x \mid x \text{ é sentença de } G\}$

1. A cadeia consiste somente de terminais
2. A cadeia pode ser derivada a partir do símbolo inicial da gramática

No Ex.1, $L(G) = \{a^n b^m;\ n>0, m>1\}$

Def5. Duas gramáticas G1 e G2 são equivalentes sse
$L(G1) = L(G2)$

Linguagens Regulares

As linguagens geradas pelas Gramáticas Regulares ou do Tipo 3 coincidem com as Linguagens Regulares (LR), também chamadas de Linguagens do Tipo 3.

Exemplo 1:

G = ({S}, {a, b}, P, S)
P = {
S -> aS
S -> b }

Exemplo 1:

$G = (\{S\}, \{a, b\}, P, S)$
$P = \{$
$S \rightarrow aS$
$S \rightarrow b \}$

Resp.: $L(G) = \{a^n b \mid n \geq 0\}$ ou a^*b

Exemplo 2:

G = ({S, A}, {a, b, c}, P, S)
P = {
S -> aS | bA
A -> c }

Exemplo 2:

G = ({S, A}, {a, b, c}, P, S)
P = {
S -> aS | bA
A -> c }
Resp.: L(G) = {a^nbc | n ≥ 0}

Exemplo 3:

G = ({<Dig>, <Int>}, {+, -, 0, ..., 9}, P, <Int>)
P = {<Int> ::= +<Dig> | -<Dig>
<Dig> ::= 0<Dig> | 1<Dig>|...| 9<Dig> | 0 | 1 | 2 |...|9 }

Exemplo 3:

G = ({<Dig>, <Int>}, {+, -, 0, ..., 9}, P, <Int>)
P = {<Int> ::= +<Dig> | -<Dig>
<Dig> ::= 0<Dig> | 1<Dig>|...| 9<Dig> | 0 | 1 | 2 |...|9 }

Resp.:
L(G) = conj. números inteiros com sinal $\pm[0..9]^+$

Exemplo 4:

$G = (\{A,B,C\}, \{0,1\}, P, A)$

$P = \{ A \to 0B \mid 0$

$B \to 1C$

$C \to 0B \mid 0 \}$

Exemplo 4:

$G = (\{A,B,C\}, \{0,1\}, P, A)$

$P = \{ A \to 0B \mid 0$

$B \to 1C$

$C \to 0B \mid 0 \}$

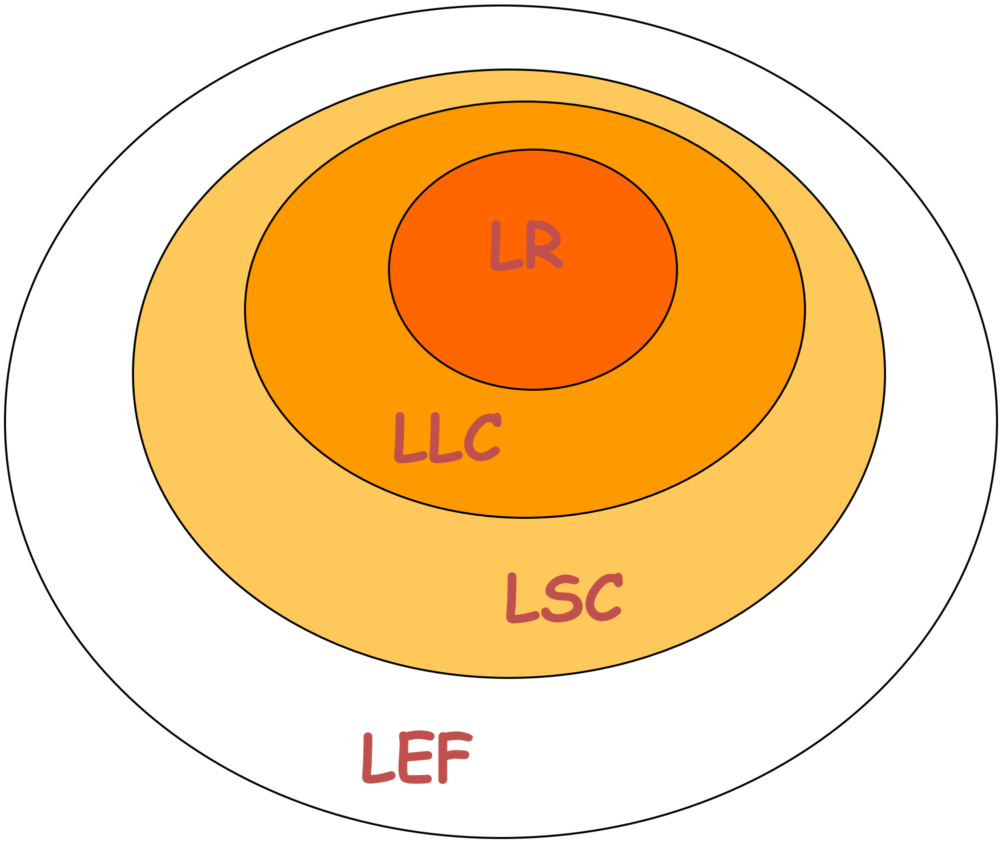
Resp.:

$L(G) = \{0(10)^*\}$

Exercícios

- Reescreva as linguagens regulares vistas anteriormente sob a forma de AF ou de Expressões Regulares, agora sob a forma de Gramáticas Regulares.

Hierarquia de Chomsky



LR = Linguagens Regulares

LLC = Linguagens Livres de Contexto

LSL =Linguagens Sensíveis ao Contexto

LEF = Linguagens com Estrutura de Frase

Gramáticas e reconhecedores

Gramáticas	**Reconhecedores**
Irrestrita	Máquina de Turing
Sensível ao contexto	Máquina de Turing com memória limitada
Livre de contexto	Autômato a pilha
Regular	Autômato finito

Linguagens e Reconhecedores

Linguagem	Gramática	Reconhecedor	Tempo para reconhecer w; |w|=n
Tipo 0: Linguagens Computáveis ou Recursivamente Enumeráveis	Gramáticas com Estrutura de Frase	Máquinas de Turing	NP-completo
Tipo 1: Sensíveis ao Contexto	Gramáticas Sensíveis ao Contexto	Máquinas de Turing com memória limitada	Exponencial: $O(2^n)$
Tipo 2: Livres de Contexto	Gramáticas Livres de Contexto	Autômatos à Pilha	Polinomial Espaço: $O(n)$; Tempo: Geral: $O(n^3)$; Não-ambíguas: $O(n^2)$; Se P= A->aB ou A->Ba ou A->a: $O(n)$
Tipo 3: Conjuntos Regulares	Gramáticas Regulares	Autômatos Finitos	Linear: $O(n)$ e $O(\|E\|)$ (no tamanho do AF)