UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Faculdade de Computação

Disciplina : Teoria da Computação

Professora : Sandra de Amo

Revisão de Gramáticas Livres do Contexto (1)

1. Fazer o exercicio 2.3 da página 128 do livro texto do Sipser, segunda edição (em ingles). Este exercicio tem resposta no livro. Tentar fazer antes de procurar a resposta !
2. Fazer o exercicio 2.4 da página 128 do livro texto do Sipser, segunda edição (em ingles). Alguns items deste exercicio tem sua resposta no livro.
3. Considere a gramática G abaixo que gera a linguagem dos palindromos sobre o alfabeto $\Sigma = \{0, 1\}$. Aplicando o algoritmo visto em aula, construa uma gramática livre do contexto na forma normal de Chomsky que seja equivalente a G.

$$\begin{array}{lcl} S & \rightarrow & \epsilon \\ S & \rightarrow & A \\ A & \rightarrow & 0A0 \\ A & \rightarrow & 1A1 \\ A & \rightarrow & \epsilon \\ A & \rightarrow & 0 \\ A & \rightarrow & 1 \end{array}$$

4. Considere as linguagens $L(G_1)$ e $L(G_2)$ geradas pelas seguintes gramáticas:

$$\begin{array}{lcl} S & \rightarrow & 0A \\ S & \rightarrow & 1A \\ A & \rightarrow & 0A \\ A & \rightarrow & 0S \\ A & \rightarrow & 1B \\ B & \rightarrow & 1B \\ B & \rightarrow & 1 \\ B & \rightarrow & 0 \\ S & \rightarrow & 0 \end{array}$$

$$
\begin{aligned}
S &\rightarrow 0A \\
A &\rightarrow 0 \\
S &\rightarrow 1B \\
B &\rightarrow 1
\end{aligned}
$$

Mostre que L(G_1) e L(G_2) são linguagens regulares (exiba um autômato correspondente a cada uma das gramáticas).

5. A partir do exercicio anterior você seria capaz de projetar um algoritmo que receba como input uma gramática G cujas regras são do tipo $A \rightarrow aB$, ou $A \rightarrow a$, onde A, B são variáveis e a é terminal, e produza como output um autômato M tal que $L(G) = L(M)$?

   Gramáticas deste tipo (equivalentes a autômatos) são chamadas de *gramáticas regulares.*

6. Mostre a inversa do exercicio anterior: projetar um algoritmo que recebe como input um autômato finito M e retorna uma gramática regular G tal que $L(G) = L(M)$.

7. Considere o alfabeto $\Sigma = \{0, 1, (,)\}$ e o conjunto das palavras sobre Σ tais que :

   (a) o número de parênteses "que abrem" é igual ao número de parênteses "que fecham".

   (b) pares de parênteses (um abrindo, outro fechando) estão apropriadamente entrelaçados. Por exemplo : (0(10)1) é uma palavra da linguagem, mas (0)10)(não é uma palavra da linguagem.

   Mostre que esta linguagem é livre do contexto exibindo uma gramática livre do contexto que a gere. Mostre que esta linguagem não é regular, aplicando o lema do bombeamento para linguagens regulares.

8. Considere a linguagem Pascal *simplificada* $\mathcal{L}_P$ definida sobre o alfabeto Σ = {:- , ; , **if**, **then**, **else**,**begin**, **end**, = , ≠} ∪ **Int** ∪ **Var** onde :

   Var = {A,B,C,D,...,Z} e **Int** = $\{-30000, \ldots, -1, 0, 1, 2, \ldots, 30000\}$.

   e cujas palavras são somente os programas constituídos por comandos de afetação de variáveis e comandos condicionais **if** (condição) **then**, **else**, onde as condições são testes do tipo $x = y$ ou $x \neq y$, onde $x \in$ **Var** e $y \in$ **Var** ou $y \in$ **Int**.

   (a) Dê alguns exemplos de palavras desta linguagem (cada palavra deve ser um programa).

9. Considere as seguintes gramáticas :

(a) $G_1 = (V_N, V_T, P, S)$ onde : $V_N = \{A, B, S\}$, $V_T = \{a,b\}$ e P é dado pelas seguintes regras :

$$\begin{array}{rcl} S & \rightarrow & Aa \\ S & \rightarrow & Ba \\ A & \rightarrow & a \\ A & \rightarrow & b \\ B & \rightarrow & a \end{array}$$

(b) $G_2 = (V_N, V_T, P, S)$ onde : $V_N = \{A, B, S\}$, $V_T = \{a,b,c\}$ e P é dado pelas seguintes regras :

$$\begin{array}{rcl} S & \rightarrow & B \\ S & \rightarrow & SABb \\ Bb & \rightarrow & bb \\ AB & \rightarrow & cc \\ B & \rightarrow & a \end{array}$$

(c) $G_3 = (V_N, V_T, P, S)$ onde : $V_N = \{A, B, S\}$, $V_T = \{a,b,c\}$ e P é dado pelas seguintes regras :

$$\begin{array}{rcl} S & \rightarrow & SAB \\ B & \rightarrow & bb \\ A & \rightarrow & aa \\ B & \rightarrow & c \end{array}$$

Em cada um destes casos, Para cada uma das gramáticas, dê duas palavras que pertencem a L(G) e duas que não pertencem.

10. Considere a seguinte gramática : $G = (V_N, V_T, P, S)$ onde : $V_N = \{A, B, S\}$, $V_T = \{a,b\}$ e P é dado pelas seguintes regras :

$$
\begin{aligned}
S &\rightarrow aB \\
S &\rightarrow bA \\
A &\rightarrow a \\
A &\rightarrow aS \\
A &\rightarrow bAA \\
B &\rightarrow b \\
B &\rightarrow bS \\
B &\rightarrow aBB
\end{aligned}
$$

Dê a árvore de derivação correspondente às seguintes palavras :

(a) ababab
(b) bbbaabaa
(c) aabbaabb

11. Considere a seguinte gramática livre do contexto :

$$
\begin{aligned}
S &\rightarrow aB \\
S &\rightarrow bA \\
A &\rightarrow a \\
A &\rightarrow aS \\
A &\rightarrow bAA \\
B &\rightarrow b \\
B &\rightarrow bS \\
B &\rightarrow aBB
\end{aligned}
$$

Dê a linguagem geradas por esta gramática. (Ver a solução na página do curso).

12. Considere as seguintes linguagens sobre o alfabeto {0,1} :

(a) conjunto das palavras que têm ao menos 3 uns.
(b) conjunto das palavras que começam e terminam com o mesmo símbolo.
(c) conjunto das palavras que têm comprimento impar.
(d) conjunto das palavras que têm comprimento impar e cujo simbolo do meio é zero.
(e) conjunto das palavras que contém mais zeros do que uns.

Dê as gramáticas que geram exatamente cada uma das linguagens acima.

13. Dê uma gramática livre do contexto gerando a seguinte linguagem :

    L = $\{w \mid w \in \{a, b\}^*$ e tal que o número de a's em w é o dobro do número de b's $\}$

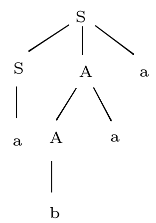
14. Considere uma gramática livre do contexto G e uma árvore de derivação $\mathcal{A}$ correspondente a uma palavra w da linguagem gerada por esta gramática.

    Mostre que, caso a gramática seja regular, só há uma maneira de construir esta árvore. Isto é, a árvore determina de forma única uma derivação $S \to \alpha_1 \to \alpha_2 \to ... \to w$.

15. Considere a gramática livre do contexto G dada pelas regras :

$$
\begin{aligned}
S &\to SAa \\
S &\to a \\
A &\to Aa \\
A &\to b
\end{aligned}
$$

    Considere a palavra $w = abaa$ pertencente a L(G) e a seguinte árvore de derivação para w :



    Mostre que existem duas maneiras de construir esta árvore, isto é existem duas sequências de regras partindo de S e chegando em w diferindo somente pela **ordem** em que as regras são aplicadas (as regras são as mesmas em cada sequência, só a ordem é que muda).

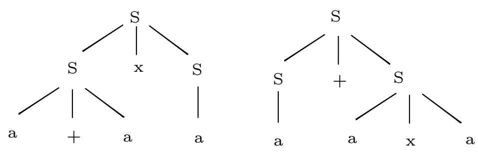
16. Considere uma árvore de derivação fixa. Uma derivação é dita *derivação mais a esquerda* se a cada nível da árvore os nós correspondentes às variáveis são desenvolvidos da esquerda para a direita. Considere o exercicio do item anterior. Qual a derivação mais a esquerda para a palavra w na árvore dada ?

    A partir de exercicio, você deve concluir que fixada uma árvore de derivação existe apenas uma derivação mais a esquerda da palavra gerada pela árvore.

17. Considere agora um outro problema. Seja a seguinte gramática livre do contexto G sobre o alfabeto $\{a, +, *, (,)\}$.

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

Considere as **DUAS** árvores de derivação seguintes para a palavra $a+a*a$



Repare que a primeira árvore corresponde a "entender" a expressão $a+a*a$ como sendo "primeiro somo $a+a$ depois multiplico o resultado por a". A segunda árvore corresponde a "entender" a expressão $a+a*a$ como sendo "somo a com o resultado da multiplicação $a*a$".

Uma gramática deste tipo, que possui **duas ou mais** árvores de derivação para uma **mesma** palavra é dita uma *gramática ambígua*. Isto porque, neste caso, uma mesma palavra w pode ter **diversos sentidos** (dependendo da árvore de derivação considerada), como mostra o exemplo acima.

Pede-se : Exiba uma gramática G' não ambígua que seja equivalente à gramática G.

Uma linguagem é dita *ambígua* se ela só pode ser gerada por gramáticas ambíguas. Assim, a linguagem gerada pela gramática G acima **NÃO É ambígua**, pois pode ser gerada por uma gramática G' não ambígua. Não confunda, portanto, linguagem ambígua com gramática ambígua.

Para o conhecimento de vocês : Existem linguagens ambíguas (veja exercício a seguir) !! Provar que uma linguagem é ambígua não é nada fácil. Por que ? O que você acha a respeito de provar que uma gramática é ambígua ?

18. **Problema muito difícil** : Considere a linguagem $L = \{a^i b^j c^k \mid i,j,k \geq 0 \text{ e } i=j \text{ ou } j=k\}$. Exiba uma gramática que gere exatamente esta linguagem. Esta gramática é ambígua ? Mostre que a linguagem L é ambígua.