

Universidade de São Paulo
Instituto de Ciências Matemáticas e de Computação
Departamento de Ciências de Computação

SCC-0605
TEORIA DA COMPUTAÇÃO E COMPILADORES

Lista de Exercícios – Capítulo 2

Linguagens e gramáticas

1. Suponha que $G = (\Sigma, V, S, P)$ seja uma GLC tal que cada produção em P ou é da forma $A \rightarrow wB$ ou da forma $A \rightarrow w$, onde $A, B \in V$ e $w \in \Sigma^*$. $L(G)$ é necessariamente uma LLD? Prove esta afirmação ou ache um contra-exemplo.
2. Converta a seguinte gramática à Forma Normal de Greibach.
$$\begin{aligned} S &\rightarrow 00A \mid B \mid 1 \\ A &\rightarrow 1AA \mid 2 \\ B &\rightarrow 0 \end{aligned}$$
3. Converta a seguinte gramática à Forma Normal de Greibach.
$$\begin{aligned} S &\rightarrow SSS \mid RS \mid 0 \\ R &\rightarrow RR \mid SR \mid 1 \end{aligned}$$
4. Escreva a gramática para a linguagem com cadeias que contenham um único a a esquerda e n b 's a direita: ab^n , $n > 0$. Qual é o tipo desta **linguagem**?
5. Construa uma gramática livre de contexto G para a linguagem $L(G) = \{ w = xayb \mid x, y \in \{a, b\}^* \}$.
6. Mostre que para qualquer linguagem livre de contexto L , $L - \{\lambda\}$ também é livre de contexto.
7. Considere uma gramática $G = (\Sigma, V, S, P)$, onde $\Sigma = \{0, 1\}$, $V = \{S\}$, $P = \{S \rightarrow 0S1, S \rightarrow 01\}$. Qual o tipo desta gramática segundo a hierarquia de Chomsky? Dê a descrição formal da linguagem gerada por esta gramática. Se for possível, descreva o autômato finito, com o menor número de estados possível, que aceite esta linguagem.
8. Considerando que:
 - toda linguagem livre de contexto também é sensível ao contexto;
 - nem toda gramática livre de contexto é sensível ao contexto;
 - há quatro tipos de gramáticas/linguagens segundo Chomsky,

as gramáticas abaixo são livres de contexto? São sensíveis ao contexto? Por que?

- a) $G = (X, V, S, P)$, onde $X = \{a, b\}$, $V = \{A, B\}$, $S = A$, $P = \{ A \rightarrow Ba, B \rightarrow BB, Aa \rightarrow Bb, B \rightarrow b, B \rightarrow bA, A \rightarrow a, Ab \rightarrow \lambda \}$
- b) $G = (X, V, S, P)$, onde $X = \{0, 1\}$, $V = \{S\}$, $P = \{S \rightarrow 0S1, S \rightarrow \lambda\}$

Que linguagem é gerada pela gramática do item b?

9. Escreva a gramática para a linguagem com cadeias que contenham um único a a esquerda e n b 's a direita: ab^n , $n > 0$. Qual é o tipo desta linguagem?
10. Construa uma gramática livre de contexto G para a linguagem $L(G) = \{w = xayb \mid x, y \in \{a, b\}^*\}$.
11. Seja $G = (\{0, 1\}, \{A, B\}, A, \{ A \rightarrow B0, B \rightarrow BB, A0 \rightarrow B1, B \rightarrow 1, B \rightarrow 1A, A \rightarrow 0, A1 \rightarrow \lambda\})$. Como você descreveria essa gramática? Essa gramática é livre de contexto? Por que? Essa gramática é sensível ao contexto? Por que?

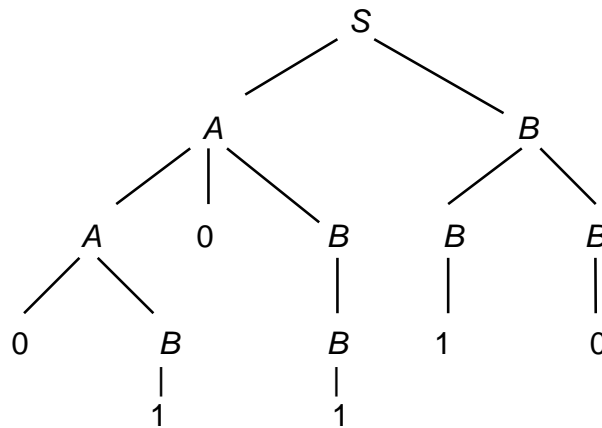
12. Construa gramáticas que gerem as seguintes linguagens:

- (a) $\{a^i b^j \mid j = i\}$
 (b) $\{a^i b^j \mid j = 2i\}$
 (c) $\{a^i b^j \mid j = i \text{ ou } j = 2i\}$
 (d) $\{a^i b^j \mid j \neq i\}$
 (e) $\{a^i b^j \mid j < i \text{ ou } j > 2i\}$
 (f) $\{a^i b^j \mid i \leq j \leq 2i\}$
 (g) $\{a^i b^j c^k \mid i = j \text{ ou } j = k \text{ ou } k = i\}$

13. Quais gramáticas são ambíguas:

- (a) $S \rightarrow bA \mid aB$
 $A \rightarrow a \mid aS \mid bAA$
 $B \rightarrow b \mid bS \mid aBB$
- (b) A gramática usada para gerar expressões aritméticas na notação posfix:
 $S \rightarrow SS+ \mid SS- \mid SS^* \mid x \mid y$
 ou na notação prefix:
 $S \rightarrow +SS \mid -SS \mid *SS \mid x \mid y$
- (c) A gramática para gerar parênteses aninhados:
 $S \rightarrow (S) \mid () \mid SS$
- (d) A gramática que define os operadores lógicos and e or
 $E \rightarrow E \text{ or } E \mid E \text{ and } E \mid (E) \mid a$

14. Suponha que G é uma gramática livre de contexto (GLC) a partir da qual a cadeia 010110 pode ser derivada. Suponha que a árvore de derivação para 010110, dada na figura abaixo, inclui todas as produções de G . Mostre que G é ambígua.



15. Observe a gramática abaixo:

$$S \rightarrow (L) \mid a$$

$$L \rightarrow L, S \mid S$$

(i) que linguagem essa gramática gera?

(ii) construa uma derivação mais à direita para $(a, (a, a))$

Autômatos de pilha

16. Dê um ACP que aceite **por pilha vazia** a linguagem gerada pela GLC:

$$S \rightarrow aAA, A \rightarrow aS \mid bS \mid a$$

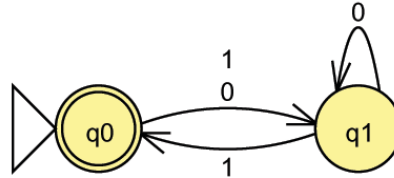
17. Dê um ACP que aceite a linguagem $\{wcw^R \mid w \in (a + b)^*\}$ **por pilha vazia**.

18. Dê um ACP que aceite a linguagem $\{ww^R \mid w \in \{0, 1\}^*\}$ **por pilha vazia**. Teste para 011110.

19. Considere a seguinte linguagem livre de contexto $L = \{0^n 1^n \mid n \geq 1\}$. Escreva um ACP M **por pilha vazia** que processe esta linguagem. Verifique como M age com as entradas 01 e 011.

20. Considere uma gramática $G = (\Sigma, V, S, P)$, onde $\Sigma = \{0, 1\}$, $V = \{S\}$, $P = \{S \rightarrow 0S1, S \rightarrow 01\}$. Qual é o ACP equivalente a esta gramática?

21. Como um ACP é constituído de uma pilha *last-in first-out* e uma máquina de estados, deve ser possível converter um autômato finito (AFD) em um ACP ignorando a pilha. Construa, se possível, um ACP a partir do seguinte AFD:



22. Considere a linguagem $L = \{ w \mid w \in (a + b)^* \text{ com número par de } a\text{'s} \}$. Por exemplo, a cadeia *abbabaa* seria aceita, enquanto que a cadeia *baabba* não.

- Se possível, escreva um ACP que processe L . Caso não seja possível, explique o porquê.
- Qual é o tipo de L ? Comente a sua resposta.

23. Considere a gramática $G = (\{a, b\}, \{S, A, B\}, S, P)$, onde P é o conjunto de produções:

$$\begin{aligned} & \{ S \rightarrow aAa \mid bBb \\ & \quad A \rightarrow b \\ & \quad B \rightarrow aA \\ & \} \end{aligned}$$

- Qual é o tipo de menor complexidade de G ?
- Qual é o tipo de menor complexidade de $L(G)$?
- Ache uma gramática na Forma Normal de Greibach para G , se possível. Se não for possível, explique o porquê.
- Ache o autômato finito que processe $L(G)$, se possível. Se não for possível, explique o porquê.
- Ache o autômato de pilha de um estado que processe $L(G)$, se possível. Se não for possível, explique o porquê.

24. Seja o seguinte autômato finito $(\{q_0, q_1\}, \{0, 1\}, \delta, q_0, \{q_0\})$:

δ	0	1
q_0	q_1	q_1
q_1	q_1	q_0

Escreva o autômato com pilha equivalente. Se não for possível, explique o porquê.

25. Seja o seguinte conjunto de produções da gramática livre de contexto G :

$$\begin{aligned} S & \rightarrow aaZcc \\ Z & \rightarrow aZc \\ Z & \rightarrow b \end{aligned}$$

- a) Qual é a linguagem que esta gramática gera?
 b) $L(G)$ é regular?

Observe agora o seguinte conjunto de produções da gramática linear à direita G' :

$$\begin{aligned} S &\rightarrow aA \\ A &\rightarrow aB \\ B &\rightarrow aB \mid bC \\ C &\rightarrow cC \mid cD \\ D &\rightarrow c \end{aligned}$$

- c) Qual é a relação entre G e G' ? São equivalentes? Por que?
 d) Escreva o autômato com pilha que processa $L(G)$.

Parsing

26. Seja o seguinte trecho de um programa em C:

```
do
{
    switch (entrada )
    {
        case 'A':
        case 'a': cont_a++; break;
    }
}
while ( int(entrada) != 27);
```

- a) construa uma gramática livre de contexto realista que gere esta cadeia;
 b) realize o parsing *top-down* da cadeia;

27. Descreva um *parse* “top-down” do texto de programa

```
begin while x ≠ y do
    begin x := 0;
        y := 0
    end
end
```

usando a gramática:

$\Sigma = \{\text{begin, end, pred, succ, :=, } \neq, \text{ while, do, }, (,), 0, x, y\}$

$V = \{C, S, S_1, S_2, A, W, U, T\}$

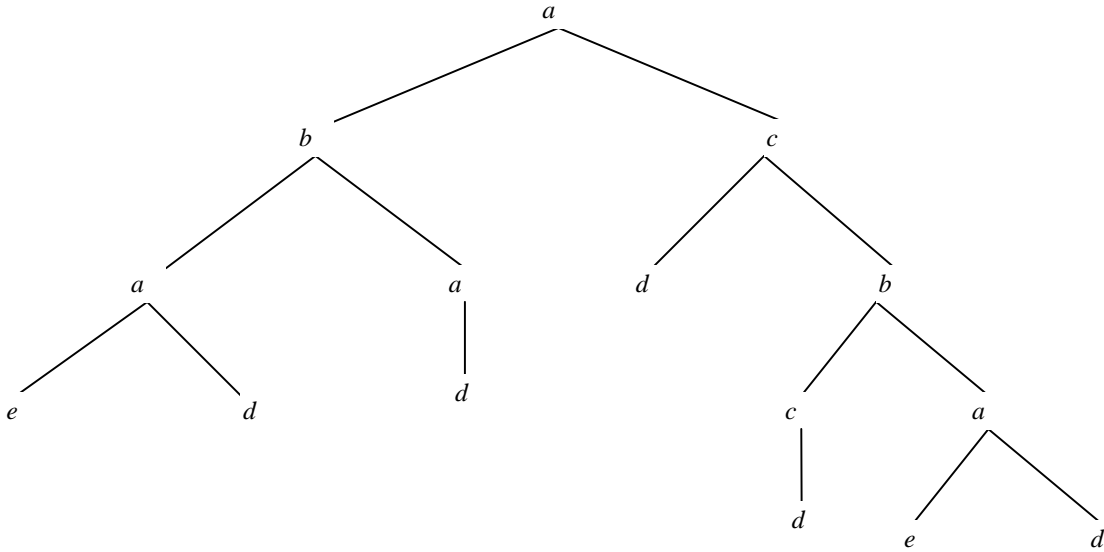
Símbolo Inicial = C

Produções =

$$\begin{aligned} C &\rightarrow \text{begin } S_1 \text{ end } \{C \text{ para comando composto}\} \\ S_1 &\rightarrow SS_2 \end{aligned}$$

$$\begin{aligned}
S_2 &\rightarrow ;S_1 \mid \lambda \\
S &\rightarrow A \mid W \mid C \\
A &\rightarrow V := T \{A \text{ para comando de atribuição}\} \\
T &\rightarrow \mathbf{pred}(V) \mid \mathbf{succ}(V) \mid 0 \\
W &\rightarrow \mathbf{while} \ V \neq V \ \mathbf{do} \ S \{W \text{ para comando while}\} \\
V &\rightarrow x \mid y
\end{aligned}$$

28. Seja a seguinte figura:



Esqueça a nossa convenção de que letras minúsculas denotam símbolos terminais e letras maiúsculas denotam variáveis. Esta figura é uma árvore de derivação de alguma gramática livre de contexto, $G = (\Sigma, V, S, P)$, para a qual as produções e símbolos não são conhecidos. Responda: **a)** Qual é a cadeia extraída desta árvore? **b)** Que símbolos devem estar em Σ ? **c)** Que símbolos estão necessariamente em V ? **d)** Você acha que d e e devem estar em Σ , ou eles poderiam estar em V ? **e)** Que produções devem estar em P ? **f)** Mostre que a cadeia $deddedd$ pertence a $L(G)$ usando o *parsing top-down*.

29. As gramáticas abaixo são LL(1)? Transforme as que não são.

- | | |
|---|--|
| <p>(a) $S \rightarrow ABc$
 $A \rightarrow a \mid \lambda$
 $B \rightarrow b \mid \lambda$</p> | <p>(c) $S \rightarrow ABBA$
 $A \rightarrow a \mid \lambda$
 $B \rightarrow b \mid \lambda$</p> |
| <p>(b) $S \rightarrow Ab$
 $A \rightarrow a \mid B \mid \lambda$
 $B \rightarrow b \mid \lambda$</p> | <p>(d) $S \rightarrow aSe \mid B$
 $B \rightarrow bBe \mid C$
 $C \rightarrow cBe \mid d$</p> |

30. Seja o seguinte trecho de um programa em C:

```

do
{
    switch (entrada )
    {
        case 'A':
        case 'a': cont_a++; break;
    }
}
while ( int(entrada) != 27);

```

- a) construa uma gramática livre de contexto realista que gere esta cadeia;
- b) realize o parsing *top-down* da cadeia;
- c) realize o parsing *bottom-up* da cadeia.

33. Considere a gramática abaixo:

$$E \rightarrow E + T \mid T$$

$$T \rightarrow T * F \mid F$$

$$F \rightarrow (E) \mid id$$

(a) Verifique se é LL(1). Se não for, transforme-a.

(b) Faça a análise sintática descendente preditiva não recursiva para a cadeia $id*id+id$

34. Quais as vantagens da separação da análise léxica da sintática?

35. Qual a diferença entre a análise sintática descendente preditiva recursiva e a não recursiva? Por que são chamadas 'preditivas'?