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

SCC-0505
INTRODUÇÃO À TEORIA DA COMPUTAÇÃO

Lista de Exercícios do Capítulo 3

Gramáticas e Linguagens Sensíveis ao Contexto

1. Seja o seguinte conjunto de produções de uma gramática livre de contexto G:

$$P = \{S \rightarrow AB,\ A \rightarrow 0,\ A \rightarrow 1,\ A \rightarrow \lambda,\ B \rightarrow 1\}$$

a) Descreva $L(G)$;
b) $L(G)$ é sensível ao contexto?
c) Se possível, ache um autômato finito que processe $L(G)$;
d) Use o Lema da Cadeia Vazia para achar a gramática equivalente sensível ao contexto.

Autômatos Limitados Linearmente e Máquinas de Turing

2. Construa uma máquina de Turing com fita limitada (ALL) que reconheça a linguagem $a^{2n}bc^{n}$, n>=1. Ilustre sua operação com o reconhecimento da sentença *aabc*.

3. Especifique um ALL que reconheça a linguagem regular 0*1+2*. Mostre os movimentos que conduzem à aceitação da cadeia 001 e à rejeição da cadeia 022.

4. Construa ALLs que reconheçam as seguintes linguagens:

a. $a(bc)^*a^* \mid a^*(b^*c \mid bc^*)a$
b. o conjunto de anagramas que podem ser obtidos a partir de uma palavra qualquer, construída sobre o alfabeto {*a*,*b*,...,*z*}, que é fornecida como entrada. Considere que a cadeia a ser analisada possui a forma palavra1-palavra2. A cadeia deve ser aceita se palavra2 for anagrama de palavra1. Caso contrário, deverá ser rejeitada
c. o conjunto das expressões aritméticas que podem ser definidas sobre o alfabeto {*a*,*b*,*c*,+,*, -,/,(,)}
d. $a^m b^n c^m d^n$, m >=1, n <> m
e. $a^m b^n c^p d^q$, m >= 1, $n > m$, $p > n$, $q > p$

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

a) Se possível, escreva um autômato limitado linearmente (ALL) que processe L. Caso não seja possível, explique o porquê.
b) Se possível, escreva um autômato de pilha (APN) que processe L. Caso não seja possível, explique o porquê.
c) Qual é o tipo de L? Comente a sua resposta.

6. Seja o seguinte conjunto de produções da gramática G:

$$S \to aSBC \mid aBC$$
$$CB \to BC$$
$$aB \to ab$$
$$bB \to bb$$
$$bC \to bc$$
$$cC \to cc$$

a) Qual o processador de linguagem de menor poder computacional capaz de processar $L(G)$ (AFN, APD, ALL ou MT)? Por que?
b) Escreva este processador.

7. Seja o seguinte conjunto de produções da gramática livre de contexto G_A:

$$S \to aaZcc$$
$$Z \to aZc$$
$$Z \to b$$

Observe agora o seguinte conjunto de produções da gramática linear a direita G_B:

$$S \to aA$$
$$A \to aB$$
$$B \to aB \mid bC$$
$$C \to cC \mid cD$$
$$D \to c$$

Qual a relação entre G_A e G_B? São equivalentes? Por que? Escreva a máquina de Turing que processa $L(G_A)$.

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

$$S \to aAa \mid bBb$$
$$A \to b$$
$$B \to aA$$

a) Ache o autômato limitado linearmente que processe $L(G)$, se possível. Se não for possível, explique o porquê.
b) Ache a máquina de Turing de uma cabeça que processe $L(G)$, se possível. Se não for possível, explique o porquê.

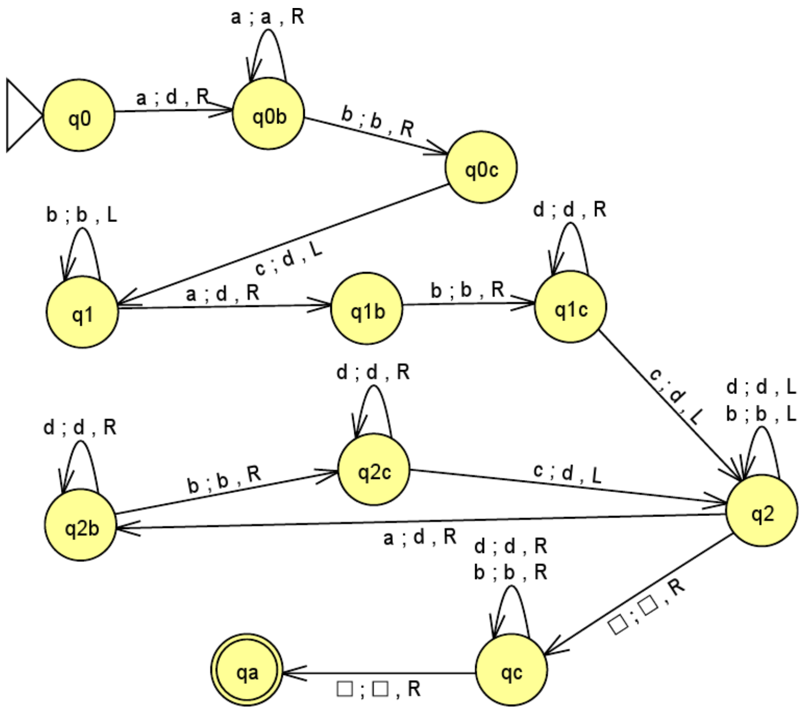
9. Seja o seguinte autômato finito $(\{q_0, q_1\}, \{0,1\}, \delta, q_0, \{q_0\})$:

δ	**0**	**1**
$\boldsymbol{q_0}$	q_1	q_1
$\boldsymbol{q_1}$	q_1	q_0

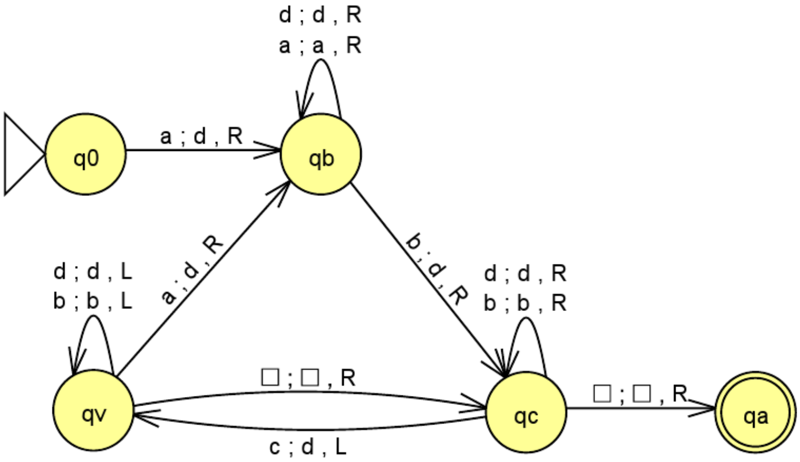
Escreva a máquina de Turing T equivalente. Se não for possível, explique o porquê.

10. Considere uma gramática $G = (\Sigma, V, S, P)$, onde $\Sigma = \{0, 1\}$, $V = \{S, A, B\}$, $P = \{S \to 0A \mid 1B \mid 0,\ A \to 0A \mid 0S \mid 1B,\ B \to 1B \mid 1 \mid 0\}$. Qual é a Máquina de Turing que processa $L(G)$?

11. Seja a Máquina de Turing M_A, representada no *JFlap*:



Seja a Máquina de Turing M_B, representada no *JFlap*:



A partir do conjunto de instruções:

a) É possível afirmar que $T(M_A)$, ou seja, o conjunto de cadeias aceitas pela Máquina de Turing M_A, é regular?
b) E quanto à $T(M_B)$?
c) Se não forem regulares, quais os tipos das linguagens processadas por M_A e por M_B?
d) Escreva os processadores de menor poder computacional que processa $T(M_A)$ e $T(M_B)$.

12. Seja T a máquina de Turing:

$T = (\{q_0, q_1, q_2, q_3\}, \{a, [,], \#\}, q_0, \{q_3\}, \delta)$

onde δ é dado por:

$\delta(q_0, a) = (q_0, a, R)$
$\delta(q_0, \#) = (q_0, \#, R)$
$\delta(q_0, [) = (q_1, \#, R)$
$\delta(q_1, [) = (q_1, [, R)$
$\delta(q_1, \#) = (q_1, \#, R)$
$\delta(q_1,]) = (q_2, \#, L)$
$\delta(q_2, x) = (q_2, x, L)$ para todo $x \neq a$
$\delta(q_2, a) = (q_0, a, R)$
$\delta(q_0, B) = (q_3, \#, R)$

Quais palavras da forma aw, onde w está em $\{[,]\}^*$, são aceitas? Você pode achar uma gramática para esta linguagem?

13. Escreva uma máquina de Turing que aceite a linguagem $(a + b)^*$, na qual há menos a's do que b's.

14. Escreva uma máquina de Turing que aceite a linguagem $(a + b)^*$ onde existe mais a's que b's.

15. Escreva uma Máquina de Turing que aceite a linguagem $(a + b)^*$, na qual há pelo menos um par de a's.

16. Sabe-se que um autômato finito (AFD/AFN) processa linguagem linear a direita (regular) e que um autômato a pilha (APN), que é equivalente a um AFN + pilha, processa linguagem livre de contexto.

Afirmação: "Qualquer máquina de Turing pode ser simulada por algum APN com duas pilhas."

Comente esta afirmação.

17. Construa a máquina de Turing que aceite o conjunto de todas as sentenças que contenham dois 0s consecutivos ou dois 1s consecutivos. Teste para 010110.

18. Considere a seguinte máquina de Turing T que reconhece a LLC $L = \{0^n1^n \mid n \geq 1\}$. Seja $T = (Q, \Sigma, q_0, q_a, \delta)$, onde

$$Q = \{q_0, q_1, \ldots, q_5\}$$
$$\Sigma = \{0, 1, Y, Z\}$$
$$q_a = q_5$$

sendo que Y e Z são símbolos da fita, mas não símbolos de entrada. δ é dado por:

1) $\delta(q_0, 0) = (q_1, Y, R)$

(T irá alternativamente substituir um 0 por Y, então um 1 por Z. No estado q_0, um 0 é substituído por um Y, e T move para a direita no estado q_1 procurando um 1.)

2) a) $\delta(q_1, 0) = (q_1, 0, R)$

b) $\delta(q_1, Z) = (q_1, Z, R)$

c) $\delta(q_1, 1) = (q_2, Z, L)$

(T se move para a direita no estado q_1 (regras 2a e 2b). Quando um 1 é encontrado, ele é mudado para um Z, e o estado se torna q_2 (regra 2c). Em q_2, vemos que T se move para a esquerda, procurando por um 0 para converter para um Y. Movendo para a esquerda, T encontrará um bloco de Z's, então talvez um bloco de 0's, então um Y.)

3) a) $\delta(q_2, Z) = (q_2, Z, L)$

b) $\delta(q_2, Y) = (q_3, Y, R)$

c) $\delta(q_2, 0) = (q_4, 0, L)$

(T se move para a esquerda através de Z's (3a). Se T encontra um Y enquanto no estado q_2, não há mais 0's para converter. T vai para o estado

q_3 para checar que não há mais 1's (3b). Se um 0 é encontrado, *T* vai para o estado q_4 e se move para a esquerda para converter o 0 mais a esquerda (3c).)

4) a) $\delta(q_4, 0) = (q_4, 0, L)$

b) $\delta(q_4, Y) = (q_0, Y, R)$

(*T* se move através de 0's (4a). Se um *Y* é encontrado, *T* passou o 0 mais a esquerda e então deve mover para a direita, para converter o 0 em um *Y*. Entra no estado q_0 e o processo descrito nas regras 1 a 4 se repete (regra 4b).)

5) a) $\delta(q_3, Z) = (q_3, Z, R)$

b) $\delta(q_3, B) = (q_5, Z, R)$

(*T* entra no estado q_3 quando não houver mais 0's (veja 3a). *T* deve mover à direita (5a). Se um branco for encontrado antes de um 1, então não há mais 1's (5b). A entrada está em *L* e *T* entra no estado q_5, o estado de aceitação.)

6) δ é indefinida, para outros casos diferentes de 1 a 5 acima.

Verifique como *T* age com a entrada 000111 através de transições entre descrições instantâneas.