Tarefas de Implementação

P1. Programa para gerar todas cadeias de comprimento l
sobre o alfabeto $\{a, b\}$

P2. Programa para gerar todas cadeias de comprimento menor que l
sobre o alfabeto $\{a, b\}$

P3. Programa para gerar todas cadeias da linguagem
$L = \{a^i b^j\}$ com i e j dados e suas permutas

P4. Programa para gerar todas cadeias de comprimento menor que l
em ordem lexicográfica

a
aa
aaa
aaaa | tam $< l$
aaab

Implementar programa para resolver o problema de correspondência de Post

- Recebe como entrada um problema de Post (conjunto de dominós)
  [b, bb], [aa, baa], [ab, a] (qualquer caractere minúsculo)

- Retorna a solução (se houver) do problema de Post
  ab|aa
  a|baa
  - avisa se não houver solução
  - avisa se não foi possível decidir

- Opção para mostrar os passos da busca
  - de forma total
  - de forma resumida: as seqüências que falharam

Lista de Exercícios I

1. Mostrar que o conjunto dos inteiros é contável.

2. Mostrar que o conjunto das funções $f: \mathbb{N} \to \mathbb{N}$ é incontável.

   Dica: considerar a função $f(n) = f_n(n) + 1$

3. Considere o alfabeto $\Sigma = \{a\}$. Descubra se Σ^* é contável.

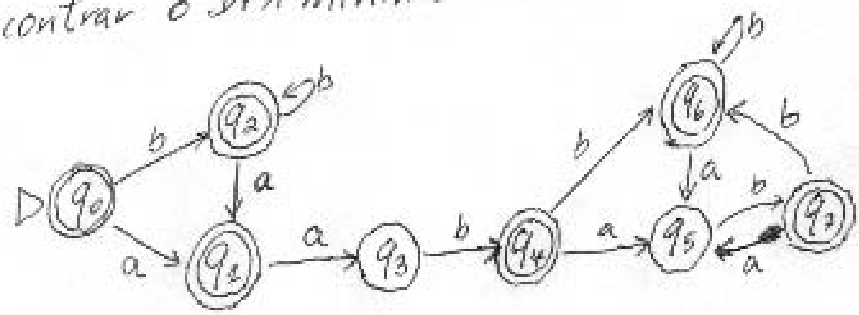
4. Considere o alfabeto $\Sigma = \{a, b\}$. Descubra se Σ^* é contável.

5. Verifique se o conjunto das cadeias de comprimento infinito sobre o alfabeto $\Sigma = \{a, b\}$ é contável.

6. Verificar se $L = \{a^n b^m a^n \mid n > 0, m > 0\}$ é uma linguagem regular.

   (Provar que existem infinitas classes de strings "indistinguíveis")

7. Encontrar o DFA mínimo



8. Encontrar o DFA mínimo que aceita apenas cadeias que contenham uma sub-cadeia "abba".

9. Encontrar o DFA mínimo que aceita apenas cadeias que não contenham nenhuma subcadeia "abba".

10. Demonstre que a intersecção de duas linguagens regulares é também linguagem regular.

Lista de Exercícios II

1. Quais as linguagens geradas pelas gramáticas G1 e G2:

$G1: S \to AB$
$A \to aA \mid a$
$B \to bB \mid \varepsilon$

$G2: S \to aS \mid aB$
$B \to bB \mid \varepsilon$

Escrever a expressão regular.

2. Para a gramática

$S \to abSc \mid A$
$A \to cAd \mid cd$

- encontrar uma árvore de derivação para ababccddcc
- escrever a linguagem gerada em notação de conjunto

3. Para a gramática

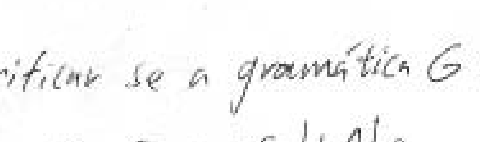
$S \to ASB \mid \varepsilon$
$A \to aAb \mid \varepsilon$
$B \to bBa \mid ba$

- derivação mais-a-esquerda
- derivação mais-a-direita
- linguagem em notação de conjunto

4. Verificar se a gramática G é regular e obter o NFA equivalente

$G: S \to aS \mid bA \mid a$
$A \to aS \mid bA \mid b$

5. Dado o NFA abaixo, encontrar gramática regular equivalente



Lista de Exercícios III

1. Sejam as gramáticas livres de contexto $G_1 = (V_1, \Sigma_1, P_1, S_1)$ e $G_2 = (V_2, \Sigma_2, P_2, S_2)$ e as linguagens geradas L_1 e L_2 respectivamente

   a) Construa uma gramática para gerar $L_1 \cup L_2$

   b) Construa uma gramática para gerar $L_1 L_2$

   c) Construa uma gramática para gerar L_1^*

   d) Verifique se as gramáticas de a, b e c são livres de contexto.

2. Mostre que a intersecção de duas linguagens livres de contexto não é necessariamente livre de contexto.

3. Utilite o Lema do Bombeamento para mostrar que a linguagem $L = \{a^i b^j a^i b^j \mid i \geq 0, j \geq 0\}$ não é livre de contexto.

4. Remover regras-ε e regras em cadeia mantendo a gramática equivalente

   a) $S \to AB \mid BCS$
   $A \to aA \mid C$
   $B \to bbB \mid b$
   $C \to cC \mid \varepsilon$

   b) $S \to aS \mid bS \mid B$
   $B \to bb \mid C \mid \varepsilon$
   $C \to cC \mid \varepsilon$

5. Remover Símbolos inúteis

   a) $S \to AA \mid CD \mid bB$
   $A \to aA \mid a$
   $B \to bB \mid bC$
   $C \to cB$
   $D \to dD \mid d$

   b) $S \to ACH \mid BB$
   $A \to aA \mid aF$
   $B \to CFH \mid b$
   $C \to aC \mid DH$
   $D \to aD \mid BD \mid Ca$
   $F \to bB \mid b$
   $H \to dH \mid d$

Lista de Exercícios - CT200 - Última - Parte 1 e 2

1. Seja $L = \{a^{2i} b^{3i} \mid i \geq 0\}$

a) Construir um PDA M_1 com $L(M_1) = L$

b) Construir um PDA atômico M_2 com $L(M_2) = L$

c) Construir um PDA estendido M_3 com $L(M_3) = L$ que tenha menos transições que M_1.

d) Traçar as computações que aceitem a string aabbb nos autômatos M_1, M_2 e M_3

PDA atômico aceita apenas transições das formas:

$$[q_j, \epsilon] \in \delta(q_i, a, \epsilon)$$
$$[q_j, \epsilon] \in \delta(q_i, \epsilon, A)$$
$$[q_j, A] \in \delta(q_i, \epsilon, \epsilon)$$

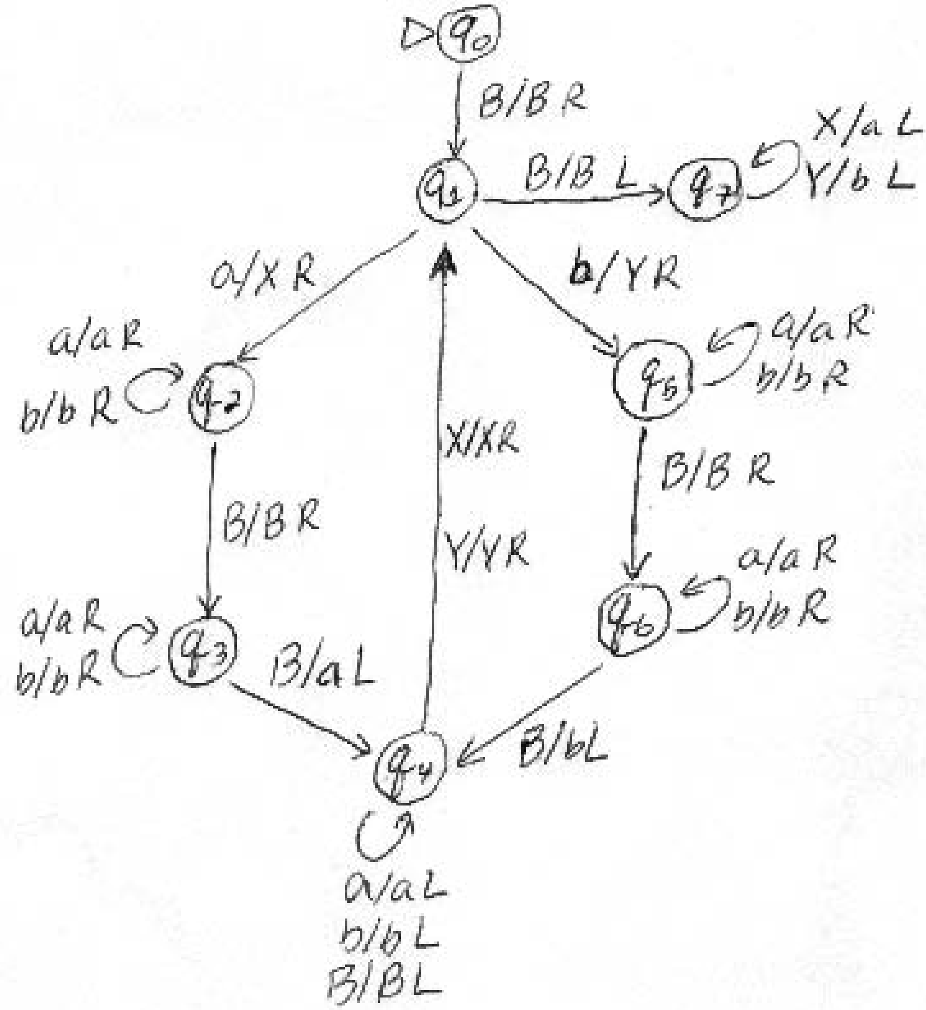
PDA estendido permite que uma única transição deposite mais de um símbolo na pilha.

2. Construir um PDA atômico que aceite a linguagem gerada pela gramática

$$S \to aABA \mid aBB$$
$$A \to bA \mid b$$
$$B \to cB \mid c$$

Lista de Exercícios – CT200 – Última – Parte 2 de 2

3. Traçar a computação da máquina de Turing abaixo quando a entrada é uma fita |B|a|a|b|B|B|...



4. Projete uma máquina de Turing que processe uma string de entrada a^i retornando sobre a própria fita a string $a^{\lfloor i/2 \rfloor}$, para qualquer $i \geq 0$. Mostre os passos da computação para entrada aaaaaBB...

5. Descreva um algoritmo para gerar a lista dos números primos. Discuta se a linguagem $\{a^i \mid i \text{ é um número primo}\}$ é recursiva e se é recursivamente enumerável.