ACH2043 - Introdução à Teoria da Computação
Lista de exercícios nº 3 (Cap. 3 Sipser)

Data para entrega: 11/12/2017

Resolução correta de 4 exercícios = nota 10.0

1) Apresente os diagramas de estados de MTs para os problemas abaixo:

a) Dada uma cadeia binária *w*, a MT deve inserir um espaço entre cada par de símbolos de *w*.

Ex: Entrada: $10010 \sqcup\sqcup \cdots$; saída: $1 \sqcup 0 \sqcup 0 \sqcup 1 \sqcup 0 \sqcup\sqcup \cdots$

b) Dado um número natural *w* em representação binária, rejeite se *w* for ímpar; se *w* for par, retorne o resultado da divisão desse número por 2.

Ex: Entrada: 10010 (34 na base 10); saída: 1001 (17 na base 10)

c) Dada uma cadeia binária *w*, escreva após a cadeia um # seguido do comprimento da cadeia (em representação binária).

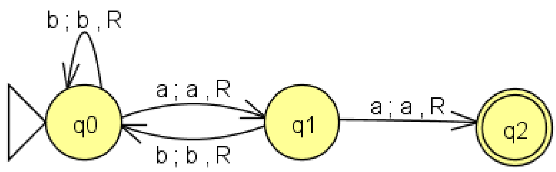
Ex: Entrada: $10010 \sqcup\sqcup \cdots$; saída: $10010\#101 \sqcup \cdots$ (note que o número após o # é 5 na base decimal)

2) Apresente descrições no nível de implementação de MTs para as linguagens abaixo, onde $\Sigma = \{0,1\}$:

a) $\{w\#x \mid w$ é uma subcadeia de $x\}$

b) $\{w \mid w$ não contém duas vezes mais 0s do que 1s$\}$

3) Dada a MT *M* abaixo, qual é a linguagem reconhecida por *M*? Apresente a expressão regular para $L(M)$ assumindo o alfabeto $\Sigma = \{a,b\}$.



Dica: Ao contrário do que ocorria se fosse um AFD, esta MT NÃO reconhece apenas cadeias terminadas em aa; se você implementar essa MT no JFlap e testar sobre algumas cadeias (terminadas ou não com aa), compreenderá por quê.

4) Mostre que a coleção de linguagens decidíveis é fechada sob concatenação. Ou seja, se duas linguagens A, B são decidíveis (com respectivos decisores M_A e M_B, respectivamente), então pode-se construir uma MT M que decide $A \circ B$. Apresente uma descrição de alto nível de M.

Dica: M deve aceitar somente cadeias $w = w_1, w_2, \ldots, w_n$ (onde n é o comprimento de w) tais que M_A aceite $w_1 \cdots w_i$ e M_B aceite $w_{i+1} \cdots w_n$ simultaneamente, para algum $i \in \{0,1,\ldots,n\}$. Lembre também que, por convenção, $w_j \cdots w_k \equiv \varepsilon$, se $k<j$.

5) Vimos em aula que, para qualquer alfabeto Σ (por definição, Σ deve ser um conjunto finito), o conjunto $S = \Sigma^+ = \Sigma\Sigma^*$ é contável, já que podemos definir um procedimento de enumeração que produz todos os seus elementos em ordem lexicográfica:

Por exemplo, para $\Sigma = \{1,2,3\}$ as cadeias de Σ^+ em ordem lexicográfica seriam:

1,2,3,11,12,13,21,22,23,31,32,33,111,112,113,121,122,123,211,212,213,...

Assumindo o alfabeto Σ = {1,2,3}, encontre uma função $f(w)$ que devolva, para cadeia *w*, seu respectivo índice na ordem lexicográfica.

Dica 1: Seja *n* o comprimento de *w*, e denote por $w_n, w_{n-1}, w_{n-2}, \ldots, w_1$ os símbolos de *w* nas posições 1,2,...,*n*, respectivamente. Apresente $f(w)$ como um polinômio sobre $w_k, w_{k-1}, w_{k-2}, \ldots, w_1$. (A inversão dos índices *n*, *n*–1 etc é uma conveniência, mas você pode usar a indexação $w_1, w_2, \ldots, w_n$ se achar mais "natural").

Dica 2: Se o alfabeto Σ fosse binário, $f(w)$ corresponderia exatamente à função de conversão de um número binário para um número decimal. Para "aquecer", você pode construir a função $f(w)$ para o alfabeto Σ = {0,1} e em seguida estendê-la para o alfabeto Σ = {1,2,3}.