Algoritmos e Fundamentos da Teoria de Computação

Lista de Exercícios 02

1 Construa uma máquina de Turing com alfabeto de entrada $\Sigma = \{a, b\}$ para aceitar cada uma das linguagens abaixo por *estado final*.

a. A linguagem formada por *strings* com o mesmo número de a's e b's (em qualquer ordem).

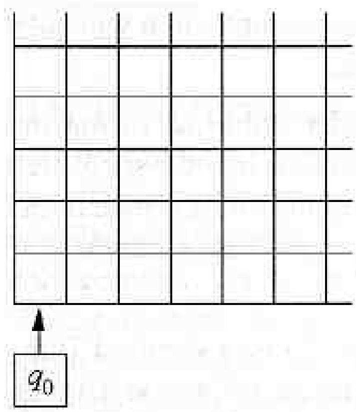
b. A linguagem formada por *strings* com um número de b's no mínimo igual ao número de a's.

2 Modifique a sua solução do exercício 1(b) para obter uma máquina de Turing que aceita a mesma linguagem por *parada*.

3 As transições de uma máquina de Turing determinística com uma fita podem ser definidas pela função parcial $\delta\colon \mathsf{Q} \times \Gamma \rightharpoonup \mathsf{Q} \times \Gamma \times \{L, R, S\}$, onde S indica que a cabeça permanece parada (*stationary*). Mostre que essa máquina é equivalente a uma máquina de Turing padrão, aonde os movimentos da cabeça são somente $\{L, R\}$.

4 Construa uma Máquina de Turing com uma fita *two-way* (isto é, infinita nos dois sentidos), com *alfabeto de entrada* $\Sigma = \{a\}$ que *para* se a fita possui pelo menos uma posição que não é branco (isto é, contendo um a). Por outro lado, se a entrada para a máquina for uma fita vazia, a computação não termina nunca. É essencial considerar que o(s) símbolo(s) a pode(m) estar em qualquer lugar da fita, não necessariamente imediatamente à direita da posição inicial da cabeça da fita. *Dica*: Lembre da prova de que os números inteiros são enumeráveis.

5 Uma máquina de Turing bidimensional é uma máquina aonde a fita consiste de um vetor de duas dimensões de quadrados, como ilustrado na figura abaixo.



Ao tomar uma transição, a máquina escreve um símbolo na posição atual da cabeça (como em todas as máquinas) e a seguir pode mover a cabeça para os quatro quadrados adjacentes (não é permitido movimento na diagonal). A computação se inicia com a cabeça lendo a posição do "canto" da fita, como mostrado na figura acima. As transições da máquina são escritas como $\delta(q_i, x) = [q_j, y, d]$, aonde d pode ser U (*up*), D (*down*), L (*left*), ou R (*right*).

Projete uma máquina de Turing bidimensional com *alfabeto de entrada* $\Sigma = \{a\}$ que *para* se a fita possui pelo menos uma posição que não é branco (isto é, contendo um a). *Obs.*: Essa é uma generalização do exercício anterior para duas dimensões. *Dica*: Lembre da prova de que os números racionais são enumeráveis.

6 Seja L a linguagem de palíndromos sobre o alfabeto $\Sigma = \{a, b\}$. Construa uma máquina de Turing de duas fitas (*two-tapes*) que aceita L, realizando no máximo $3n + 4$ transições, onde n é o tamanho da entrada.

7 Construa uma máquina de Turing de duas fitas (*two-tapes*) com *alfabeto de entrada* $\Sigma = \{a, b\}$ que aceita a linguagem $\{a^i b^{2i} \mid i \geq 0\}$. A cabeça de leitura da fita 1, que contém a entrada, deve se mover somente da esquerda para a direita, isto é, a entrada só pode ser lida uma vez (*one pass*).

8 Construa uma máquina de Turing não-determinística cuja linguagem é o conjunto de *strings* sobre o alfabeto $\Sigma = \{a, b\}$ que contém uma *substring* u que satisfaz as seguintes propriedades:

i) o tamanho de u é maior ou igual a três;

ii) u possui o mesmo número de a's e b's.

Assuma a existência de uma máquina determinística M que aceita uma *string* u se ela satisfaz a propriedade ii) e rejeita u, caso contrário. (Veja exercício 1(a).) Use M para construir a máquina não-determinística pedida.

9 Construa uma máquina de Turing não-determinística com *duas fitas* cuja linguagem é o conjunto de *strings* sobre o alfabeto $\Sigma = \{a, b\}$, definida como $\mathsf{L} = \{uu \mid u \in \Sigma^*\}$. Qualquer computação para uma entrada w deve terminar com a máquina realizando no máximo $2n + 2$ transições, onde n é o tamanho da entrada w.