MAC 4722 - Linguagens, Autômatos e Computabilidade

Rodrigo Augusto Dias Faria - NUSP 9374992
Departamento de Ciência da Computação - IME/USP

30 de maio de 2016

Lista 6

L6.1 (Sipser 3.06) No Teorema 3.21 mostramos que uma linguagem é Turing-reconhecível sse algum enumerador a enumera. Por que não usamos o seguinte algoritmo mais simples para a direção de ida da prova? Tal qual anteriormente, $s_1, s_2, \ldots$ é uma lista de todas as cadeias em Σ^*.

$E =$ "Ignore a entrada.

1. Repita o que se segue para $i = 1, 2, 3, \ldots$

2. Rode M sobre s_i.

3. Se ela aceita, imprima s_i."

Resposta: O algoritmo indica que devemos rodar M sobre todas as cadeias possíveis de Σ^*. A mudança sugerida no enunciado faz com que M execute em uma cadeia s_i por vez e, caso M aceite s_i, o enumerador E imprime-na. Ocorre que M pode entrar em *loop* para uma cadeia s_k qualquer e, por conseguinte, nenhuma outra cadeia subsequente a s_k será impressa por E e, portanto, a linguagem de M será diferente do conjunto de cadeias listadas por E.

L6.2 (Sipser 3.08) Dê descrições a nível de implementação de máquinas de Turing que decidem as linguagens abaixo sobre o alfabeto $\{0, 1\}$.

a. $\{w \mid w$ contém o mesmo número de 0s e 1s$\}$

Resposta: Vamos chamar de M_a a MT que decide a linguagem em **a.** Note que o primeiro passo deve aceitar a cadeia vazia pois, neste caso, o número de 0s e 1s é igual a zero.

$M_a =$ "Sobre a cadeia de entrada w:

1. Se w é a cadeia vazia, *aceite*, caso contrário, vá para o passo 2.

2. Faça uma varredura em w da esquerda para a direita e marque o primeiro 0 que ainda não esteja marcado. Se nenhum 0 desmarcado foi encontrado, *rejeite*. Retorne a cabeça para a extremidade esquerda da fita.

3. Faça uma varredura em w da esquerda para a direita e marque o primeiro 1 que ainda não esteja marcado. Se a varredura não encontrou nenhum 1 desmarcado, *rejeite*. Retorne a cabeça para a extremidade esquerda da fita.

4. Faça uma nova varredura em w da esquerda para a direita. Se um 0 desmarcado for encontrado, volte a cabeça uma posição à esquerda e retorne ao passo 2, caso contrário, mova a cabeça para a extremidade esquerda da fita e passe ao passo 5.

5. Novamente, faça uma varredura em w da esquerda para a direita. Se houver um 1 desmarcado, *rejeite*, senão *aceite*.

b. $\{w \mid w$ contém duas vezes mais 0s que 1s$\}$

Resposta: Vamos chamar de M_b a MT que decide a linguagem em **b.** A estratégia utilizada aqui é similar à linguagem do item **a.**

$M_b =$ "Sobre a cadeia de entrada w:

1. Faça uma varredura em w da esquerda para a direita e marque o primeiro 0 que ainda não esteja marcado. Mova a cabeça para a direita até encontrar o segundo 0 que não esteja marcado e marque-o. Se nenhum ou apenas um 0 desmarcado foi encontrado, *rejeite*. Retorne a cabeça para a extremidade esquerda da fita.

2. Faça uma varredura em w da esquerda para a direita e marque o primeiro 1 que ainda não esteja marcado. Se a varredura não encontrou nenhum 1 desmarcado, *rejeite*. Retorne a cabeça para a extremidade esquerda da fita.

3. Faça uma nova varredura em w da esquerda para a direita. Se pelo menos dois 0s desmarcados foram encontrados, retorne ao passo 1, caso contrário, passe ao passo 4. Antes de mover-se para o passo decidido, mova a cabeça para a extremidade esquerda da fita.

4. Novamente, faça uma varredura em w da esquerda para a direita. Se houver um 1 desmarcado, *rejeite*, senão *aceite*.

c. $\{w \mid w$ não contém duas vezes mais 0s que 1s$\}$

Resposta: Vamos chamar de M_c a MT que decide a linguagem em **c.** A estratégia utilizada aqui é aplicar a MT obtida em **b.** como uma subrotina.

$M_c =$ "Sobre a cadeia de entrada w:

1. Rode w na máquina M_b.

2. Se M_b aceita w, *rejeite*, senão *aceite*.

L6.3 (Sipser 3.14) Um ***autômato com fila*** é como um autômato com pilha, exceto que a pilha é substituída por uma fila. Uma ***fila*** é uma fita que permite que símbolos sejam escritos somente na extremidade esquerda e lidos somente da extremidade direita. Cada operação de escrita (denominá-la-emos *empurrar*) adiciona um símbolo na extremidade esquerda da fila e cada operação de leitura (denominá-la-emos *puxar*) lê e remove um símbolo na extremidade direita. Como com um `AP`, a entrada é colocada numa fita de entrada de somente-leitura separada, e a cabeça sobre a fita de entrada pode mover somente da esquerda para a direita. A fita de entrada contem uma célula com um símbolo em branco após a entrada, de modo que essa extremidade da entrada possa ser detectada. Um autômato com fila aceita sua entrada entrando num estado especial de aceitação em qualquer momento. Mostre que uma linguagem pode ser reconhecida por um autômato com fila determinístico **sse** a linguagem é Turing-reconhecível.

Resposta: TODO

L6.4 (Sipser 3.16) Mostre que a coleção de linguagens Turing-reconhecíveis é fechada sob a operação de

a. união

Resposta: Seja L_1 e L_2 linguagens Turing-reconhecíveis. Logo, existem máquinas de Turing M_1 e M_2 que reconhecem L_1 e L_2, respectivamente. Vamos construir uma máquina de Turing M capaz de reconhecer a união $L_1 \cup L_2$ da seguinte forma:

$M =$ "Sobre a cadeia de entrada w:

1. Repita o seguinte para $i = 1, 2, 3, \ldots$

2. Rode as máquinas M_1 e M_2 sobre w por i passos. Se M_1 ou M_2 aceita w, *aceite*. Se M_1 e M_2 param e rejeitam w, então *rejeite*."

A máquina construída reconhece cadeias de $L_1 \cup L_2$, pois se M_1 ou M_2 aceitam w, em algum momento a máquina M aceitará w, já que a máquina vai, passo a passo, tentando reconhecer w em M_1 e M_2 simultaneamente. M poderá entrar em *loop* se M_1 ou M_2 entrar em *loop* e ambas rejeitarem w.

b. concatenação

Resposta: Seja L_1 e L_2 linguagens Turing-reconhecíveis. Logo, existem máquinas de Turing M_1 e M_2 que reconhecem L_1 e L_2, respectivamente. Vamos construir uma máquina de Turing M capaz de reconhecer a concatenação L_1L_2.

c. estrela

d. intersecção

Resposta: Seja L_1 e L_2 linguagens Turing-reconhecíveis. Logo, existem máquinas de Turing M_1 e M_2 que reconhecem L_1 e L_2, respectivamente. Vamos construir uma máquina de Turing M capaz de reconhecer a intersecção $L_1 \cap L_2$ da seguinte forma:

$M =$ "Sobre a cadeia de entrada w:

1. Repita o seguinte para $i = 1, 2, 3, \ldots$

2. Rode as máquinas M_1 e M_2 sobre w por i passos. Se M_1 e M_2 aceitam w, *aceite*. Se M_1 ou M_2 para e rejeita w, então *rejeita*."

A máquina construída aceita cadeias de $L_1 \cap L_2$, pois se $w \in L_1 \cap L_2$, significa que $\exists w \mid w \in L_1$ e $w \in L_2$. Como a máquina testa w em M_1 e M_2, e só aceita caso w seja aceita por ambas, temos que a máquina M reconhece as cadeias de $L_1 \cap L_2$. M poderá entrar em *loop* se M_1 ou M_2 entrar em *loop* e rejeitar w.