Introdução à Teoria da Computação Exercícios

Livro: Michel Sipser, Introdução à Teoria da Computação – 2ª Ed. – Capítulo 03

EXERCÍCIOS

3.1 Este exercício concerne à MT M_2 cuja descrição e diagrama de estados aparecem no Exemplo 3.7. Em cada um dos itens a seguir, dê a seqüência de configurações nas quais M_2 entra quando iniciada sobre a cadeia de entrada indicada.

a. 0.

R**b.** 00.

c. 000.

d. 000000.

3.2 Este exercício concerne à MT M_1 cuja descrição e diagrama de estados aparecem no Exemplo 3.9. Em cada um dos itens a seguir, dê a seqüência de configurações nas quais M_1 entra quando iniciada sobre a cadeia de entrada indicada.

R**a.** 11.

b. 1#1.

c. 1##1.

d. 10#11.

e. 10#10.

R**3.3** Modifique a prova do Teorema 3.16 para obter o Corolário 3.19, mostrando que uma linguagem é decidível sse alguma máquina de Turing não-determinística a decide. (Você pode assumir o teorema seguinte sobre árvores. Se todo nó em uma árvore tem uma quantidade finita de filhos e todo ramo da árvore tem uma quantidade finita de nós, a árvore propriamente dita tem uma quantidade finita de nós.)

3.4 Dê uma definição formal de um enumerador. Considere-o como um tipo de máquina de Turing de duas fitas que usa sua segunda fita como a impressora. Inclua uma definição da linguagem enumerada.

R**3.5** Examine a definição formal de uma máquina de Turing para responder às seguintes perguntas e explique seu raciocínio.

a. Uma máquina de Turing pode alguma vez escrever o símbolo branco ␣ em sua fita?

b. O alfabeto de fita Γ pode ser o mesmo que o alfabeto de entrada Σ?

c. A cabeça de uma máquina de Turing pode *alguma vez* estar na mesma localização em dois passos sucessivos?

d. Uma máquina de Turing pode conter apenas um único estado?

3.6 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."

3.7 Explique por que a descrição a seguir não é uma descrição de uma máquina de Turing legítima.

M_{ruim} = "A entrada é um polinômio p sobre as variáveis $x_1, \ldots, x_k$.
1. Tente todas as possíveis valorações de $x_1, \ldots, x_k$ para valores inteiros.
2. Calcule o valor de p sobre todas essas valorações.
3. Se alguma dessas valorações torna o valor de p igual a 0, *aceite*; caso contrário, *rejeite*."

3.8 Dê descrições no nível de implementação de máquinas de Turing que decidam as seguintes linguagens sobre o alfabeto {0,1}.

R**a.** $\{w|\ w$ possui o mesmo número de 0s e 1s$\}$
b. $\{w|\ w$ contém duas vezes mais 0s que 1s$\}$
c. $\{w|\ w$ não contém duas vezes mais 0s que 1s$\}$

PROBLEMAS

3.9 Seja um k-AP um autômato com pilha que tem k pilhas. Portanto, um 0-AP é um AFN e um 1-AP é um AP convencional. Você já sabe que 1-APs são mais poderosos (reconhecem uma classe maior de linguagens) que 0-APs.

a. Mostre que 2-APs são mais poderosos que 1-APs.
b. Mostre que 3-APs não são mais poderosos que 2-APs.
(Dica: Simule uma fita de máquina de Turing com duas pilhas.)

R**3.10** Digamos que uma ***máquina de Turing de escrita-única*** é uma MT de uma única fita que pode alterar cada célula de fita no máximo uma vez (incluindo a parte da entrada da fita). Mostre que essa variante do modelo da máquina de Turing é equivalente ao modelo comum da máquina de Turing. (Dica: Como um primeiro passo, considere o caso no qual a máquina de Turing pode alterar cada célula de fita no máximo duas vezes. Use bastante fita.)

3.11 Uma ***máquina de Turing com fita duplamente infinita*** é semelhante a uma máquina de Turing comum, mas sua fita é infinita para a esquerda assim como para a direita. A fita é inicialmente preenchida com brancos com exceção da parte que contém a entrada. A computação é definida como de costume, exceto que a cabeça nunca encontra um final da fita à medida que ela move para a esquerda. Mostre que esse tipo de máquina de Turing reconhece a classe de linguagens Turing-reconhecíveis.

3.12 Uma ***máquina de Turing com reinicialização à esquerda*** é semelhante a uma máquina de Turing comum, mas a função de transição tem a forma

$$\delta\colon Q \times \Gamma \longrightarrow Q \times \Gamma \times \{\text{D}, \text{REINICIA}\}.$$

Se $\delta(q, a) = (r, b, \text{REINICIA})$, quando a máquina está no estado q lendo um a, a sua cabeça salta para a extremidade esquerda da fita depois que ela escreve b na fita e entra no estado r. Note que essas máquinas não têm a capacidade usual de mover a cabeça um símbolo para a esquerda. Mostre que máquinas de Turing com reinicialização à esquerda reconhecem a classe de linguagens Turing-reconhecíveis.

3.13 Uma ***máquina de Turing com movimento nulo em vez de à esquerda*** é semelhante a uma máquina de Turing comum, mas a função de transição tem a forma

$$\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{\mathrm{D}, \mathrm{P}\}.$$

Em cada ponto a máquina pode mover sua cabeça para a direita ou deixá-la parada na mesma posição. Mostre que essa variante da máquina de Turing *não* é equivalente à versão usual. Que classe de linguagens essas máquinas reconhecem?

3.14 Um ***autômato com fila*** é como um 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 contém 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.

3.15 Mostre que a coleção de linguagens decidíveis é fechada sob a operação de

- R**a.** união.
- **b.** concatenação.
- **c.** estrela.
- **d.** complementação.
- **e.** interseção.

3.16 Mostre que a coleção de linguagens Turing-reconhecíveis é fechada sob a operação de

- R**a.** união.
- **b.** concatenação.
- **c.** estrela.
- **d.** interseção.

***3.17** Seja $B = \{\langle M_1 \rangle, \langle M_2 \rangle, \ldots\}$ uma linguagem Turing-reconhecível constituída de descrições de MTs. Mostre que existe uma linguagem decidível C também consistindo de descrições de MTs tal que toda máquina descrita em B tem uma máquina equivalente em C e vice-versa.

***3.18** Mostre que uma linguagem é decidível sse algum enumerador enumera a linguagem em ordem lexicográfica.

***3.19** Mostre que toda linguagem Turing-reconhecível infinita tem um subconjunto infinito decidível.

***3.20** Mostre que MTs de uma única fita que não podem escrever na parte da fita contendo a cadeia de entrada reconhecem somente linguagens regulares.

3.21 Seja $c_1x^n + c_2x^{n-1} + \cdots + c_nx + c_{n+1}$ um polinômio com uma raiz em $x = x_0$. Suponha que $c_{\text{máx}}$ seja o maior valor absoluto de um c_i. Mostre que

$$|x_0| < (n+1)\frac{c_{\text{máx}}}{|c_1|}.$$

R**3.22** Seja A a linguagem contendo uma única cadeia, s, onde

$$s = \begin{cases} 0 & \text{se vida nunca será encontrada em Marte.} \\ 1 & \text{se vida será encontrada em Marte algum dia.} \end{cases}$$

A é decidível? Por que sim ou por que não? Para os propósitos deste problema, assuma que a questão de se vida será encontrada em Marte tem uma resposta não ambígua SIM ou NÃO.

SOLUÇÕES SELECIONADAS

3.1 **(b)** $q_1 00$, $\sqcup q_2 0$, $\sqcup \mathtt{x} q_3 \sqcup$, $\sqcup q_5 \mathtt{x} \sqcup$, $q_5 \sqcup \mathtt{x} \sqcup$, $\sqcup q_2 \mathtt{x} \sqcup$, $\sqcup \mathtt{x} q_2 \sqcup$, $\sqcup \mathtt{x} \sqcup q_{\text{aceita}}$

3.2 **(a)** $q_1 11$, $\mathtt{x} q_3 1$, $\mathtt{x}1 q_3 \sqcup$, $\mathtt{x}1\sqcup q_{\text{rejeita}}$.

3.3 Provamos ambas as direções do "sse." Primeiro, se uma linguagem L for decidível, ela pode ser decidida por uma máquina de Turing determinística, e essa é automaticamente uma máquina de Turing não-determinística.

Segundo, se uma linguagem L for decidida por uma MT não-determinística N, construímos uma MT determinística D_2 que decide L. A máquina D_2 roda o mesmo algoritmo que aparece na MT D descrita na prova do Teorema 3.16, com um Estágio 5 adicional: *Rejeite* se todos os ramos do não-determinismo de N estão esgotados.

Argumentamos que D_2 é um decisor para L. Se N aceita sua entrada, D_2 em algum momento no futuro encontrará um ramo de aceitação e aceitará também. Se N rejeita sua entrada, todos os seus ramos param e rejeitam porque ela é um decisor. Logo, cada um dos ramos tem uma quantidade finita de nós, onde cada nó representa um passo da computação de N ao longo daquele ramo. Conseqüentemente, a árvore inteira da computação de N sobre essa entrada é finita, em virtude do teorema sobre árvores dado no enunciado do exercício. Portanto, D vai parar e rejeitar quando essa árvore inteira tiver sido explorada.

3.5 **(a)** Sim. O alfabeto de fita Γ contém $\sqcup$. Uma máquina de Turing pode escrever quaisquer caracteres em Γ na sua fita.

(b) Não. Σ nunca contém $\sqcup$, mas Γ sempre contém $\sqcup$. Portanto, eles nunca podem ser iguais.

(c) Sim. Se a máquina de Turing tenta mover sua cabeça para a esquerda da extremidade esquerda, ela permanece na mesma célula da fita.

(d) Não. Qualquer máquina de Turing tem que conter dois estados distintos q_{aceita} e q_{rejeita}. Portanto, uma máquina de Turing contém pelo menos dois estados.

3.8 **(a)** "Sobre a cadeia de entrada w:

1. Faça uma varredura na fita e marque o primeiro 0 que não foi marcado. Se nenhum 0 não marcado for encontrado, vá para o estágio 4. Caso contrário, mova a cabeça de volta para a frente da fita.

2. Faça uma varredura na fita e marque o primeiro 1 que não tiver sido marcado. Se nenhum 1 não marcado for encontrado, *rejeite*.

3. Mova a cabeça de volta para a frente da fita e vá para o estágio 1.

4. Mova a cabeça de volta para a frente da fita. Faça uma varredura na fita para ver se ainda resta algum 1 não marcado. Se nenhum for encontrado, *aceite*; caso contrário, *rejeite*."

3.10 Primeiro simulamos uma máquina de Turing comum por uma máquina de Turing escreve-duas-vezes. A máquina de Turing escreve-duas-vezes simula um único passo da máquina original copiando a fita inteira para uma parte nova da fita do lado direito da parte correntemente utilizada. O procedimento de cópia opera caracter a caracter, marcando um caracter à medida que ele é copiado. Esse procedimento altera cada célula de fita duas vezes, uma vez para escrever o caracter pela primeira vez e novamente para marcar que ele foi copiado. A posição da cabeça da máquina de Turing original é marcada na fita. Durante a cópia das células, na posição marcada ou em posições adjacentes, o conteúdo da fita é atualizado conforme as regras da máquina de Turing original.

Para realizar a simulação com uma máquina de escrita-única, opere como antes, exceto que cada célula da fita anterior é agora representada por duas células. A primeira delas contém o símbolo de fita da máquina original e a segunda é para a marca usada no procedimento de cópia. A entrada não é apresentada à máquina no formato com duas células por símbolo, portanto na primeira vez que a fita é copiada, as marcas de cópia são colocadas diretamente sobre os símbolos de entrada.

3.15 **(a)** Para quaisquer duas linguagens decidíveis L_1 e L_2, sejam M_1 e M_2 MTs que as decidem. Construímos uma MT M' que decide a união de L_1 e L_2:

"Sobre a entrada w:

1. Rode M_1 sobre w. Se ela aceita, *aceite*.

2. Rode M_2 sobre w. Se ela aceita, *aceite*. Caso contrário, *rejeite*."

M' aceita w se M_1 ou M_2 a aceita. Se ambas rejeitam, M' rejeita.

3.16 **(a)** Para quaisquer duas linguagens Turing-reconhecíveis L_1 e L_2, sejam M_1 e M_2 MTs que as reconhecem. Construímos uma MT M' que reconhece a união de L_1 e L_2:

"Sobre a entrada w:

1. Rode M_1 e M_2 alternadamente sobre w passo a passo. Se alguma aceita, *aceite*. Se ambas param e rejeitam, *rejeite*."

Se M_1 ou M_2 aceitam w, M' aceita w porque a MT que aceita chega a seu estado de aceitação após um número finito de passos. Note que se ambas M_1 e M_2 rejeitam e uma delas faz isso entrando em loop, então M' vai entrar em loop.

3.22 A linguagem A é uma das duas linguagens, $\{0\}$ ou $\{1\}$. Em qualquer dos casos a linguagem é finita, e, portanto, decidível. Se você não é capaz de determinar qual dessas duas linguagens é A, você não será capaz de descrever o decisor para A, mas você pode dar duas máquinas de Turing, uma das quais é o decisor de A.