Máquinas de Turing e variantes

IBM1088 Linguagens Formais e Teoria da Computação

Evandro Eduardo Seron Ruiz
evandro@usp.br

Departmento de Computação e Matemática
FFCLRP
Universidade de São Paulo

Homenagem do dia

Alan Turing

If a machine is expected to be infallible, it cannot also be intelligent.

AT: Matemático, Criptoanalista e Cientista da Computação Inglês, 1912–1954.

Prólogo

- Máquina de Turing é uma formalização da 'máquina universal'
- λ-cálculo é outra
- Existem variações da MT (controle, fita, etc.)
- Detalharemos uma MT como autômato e
- Veremos algumas variações da MT

Conteúdo

1. Recordação
2. Representação da MT
3. MT e linguagens
4. Variantes da MT
   - MT multifitas
   - MT não determinística
5. Exercício em sala

1. Recordação
2. Representação da MT
3. MT e linguagens
4. Variantes da MT
   - MT multifitas
   - MT não determinística
5. Exercício em sala

Turing-reconhecível

Linguagens recursivamente enumeráveis

Chama-se de uma linguagem Turing-reconhecível se alguma MT a reconhece.

- São as Linguagens recursivamente enumeráveis.
- As linguagens reconhecidas mecanicamente
- Nas linguagens recursivamente enumeráveis a MT pode parar num estado não final ou entrar em loop

Turing-decidível

Linguagens recursivas

Chama-se de uma linguagem Turing-decidível, ou simplesmente dedidível, se alguma MT a decide.

Turing-decidível

Linguagens recursivas

Chama-se de uma linguagem Turing-decidível, ou simplesmente dedidível, se alguma MT a decide.

Toda linguagem decidível é Turing-reconhecível. O inverso não se aplica.

Turing-decidível

Linguagens recursivas

Chama-se de uma linguagem Turing-decidível, ou simplesmente dedidível, se alguma MT a decide.

Toda linguagem decidível é Turing-reconhecível. O inverso não se aplica.

Pergunta

Existem linguagens sobre $\Sigma = \{0, 1\}$ que não são decidíveis?

Sim, existem problemas não solúveis computacionalmente.

1 Recordação

2 Representação da MT

3 MT e linguagens

4 Variantes da MT
- MT multifitas
- MT não determinística

5 Exercício em sala

Representação da MT

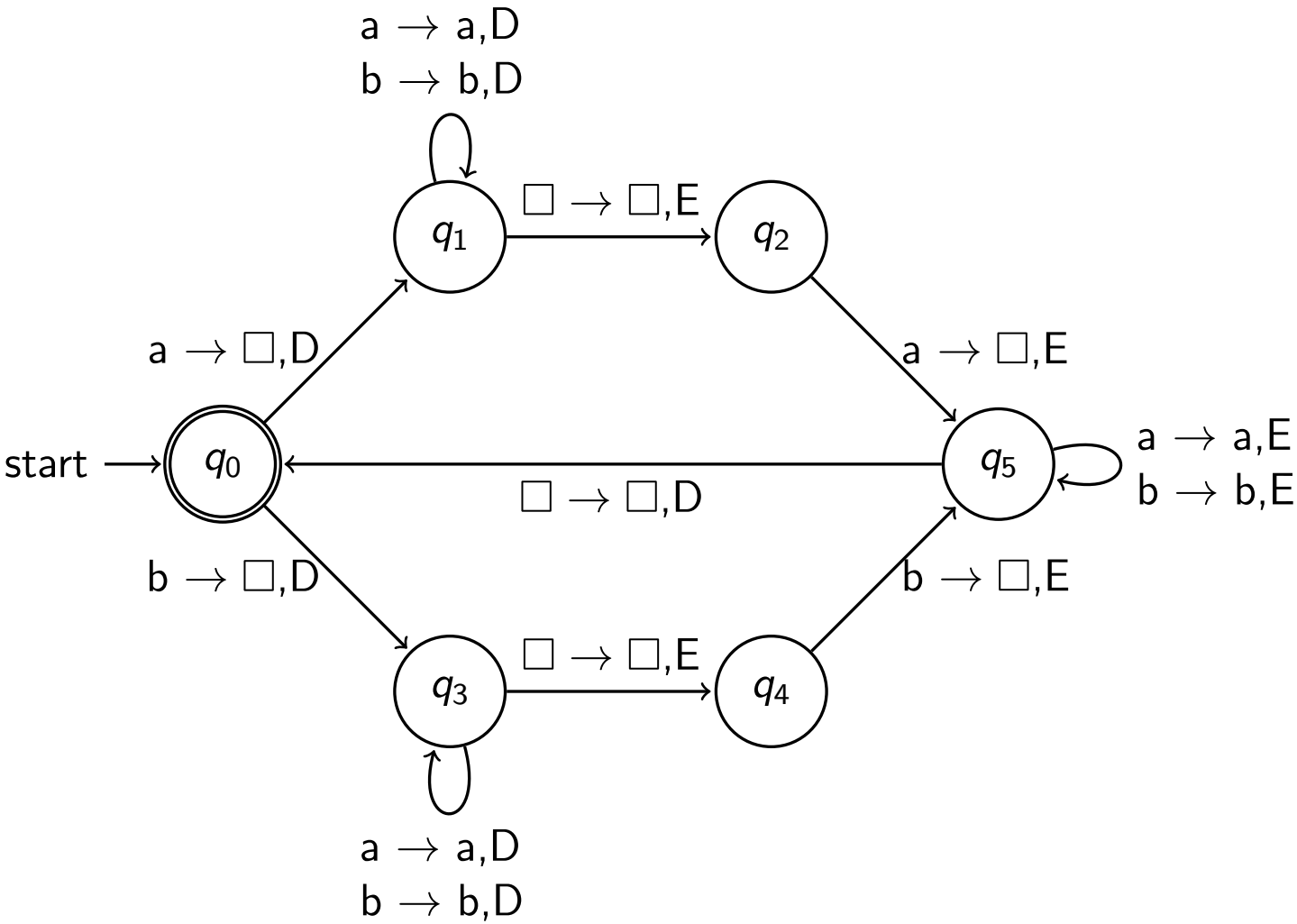
- Para facilitar a leitura, o espaço em branco '␣' será grafado como □
- No exemplo abaixo, na transição q_0 para q_1 o rótulo a $\rightarrow$ □,D significa 'ler a, apagá-lo e mover a cabeça de leitura para direita'
- Formalmente: $\delta(q_0, a) \rightarrow (q_1, \square, D)$, ou
- $\delta(q_0, a) \rightarrow (q_1, \text{␣}, D)$

Vejamos um exemplo:

Exemplo de MT

- MT que reconhece palíndromes pares em $\Sigma = \{a, b\}$, e.g., aabbbbaa (é uma LLC)
- A cadeia deve ler lida da esquerda para direita
- O autômato começa sobrescrevendo um símbolo em branco ($\square$) no primeiro símbolo, e.g, 'a', para saber onde retornar (início da cadeia)
- Após ler o primeiro 'a' e este ser apagado, a máquina avança por todos os caracteres sem alterá-los
- Chegando ao final da cadeia, ($\square$), a cabeça lê o caractere correspondente ao primeiro e avança à esquerda
- Então passa por todos os demais caracteres sem alterá-los, e
- Chegando ao início da cadeia ($\square$), move-se para a direita

Exemplo de MT para reconhecer palíndromes pares



Observação

Sobre a última MT

Percebam que neste exemplo os estados de rejeição não estão ilustrados por uma questão de minimização da máquina.
Sabemos que em qualquer estado da MT, caso seja lido um símbolo não esperado, a MT rejeita a cadeia.

1 Recordação

2 Representação da MT

3 MT e linguagens

4 Variantes da MT
- MT multifitas
- MT não determinística

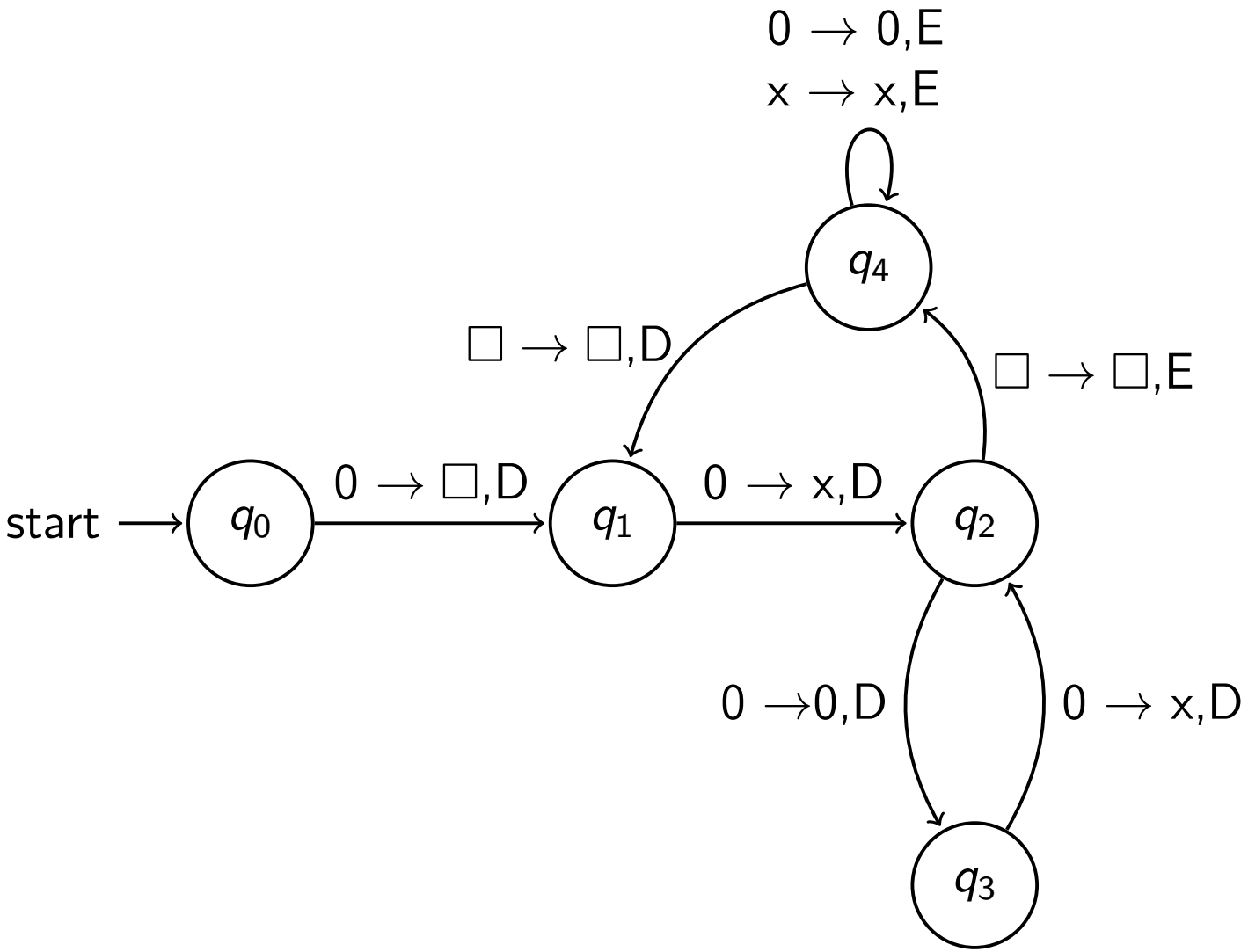
5 Exercício em sala

Explicando o reconhecedor de $L_2 = \{0^{2^n} \mid n \geq 0\}$ (1/2)

- Marca o primeiro zero com $\square$ para delimitar o início da cadeia
- A cada dois '0', marca o segundo com 'x' e move a direita
- Quando chegar ao final da cadeia, volta ao seu início

Agora vamos completar o autômato acrescentando um nó de rejeição (q_R) e um de aceitação (q_A)

MT para reconhecer $L_2 = \{0^{2^n} \mid n \geq 0\}$ (1/2)

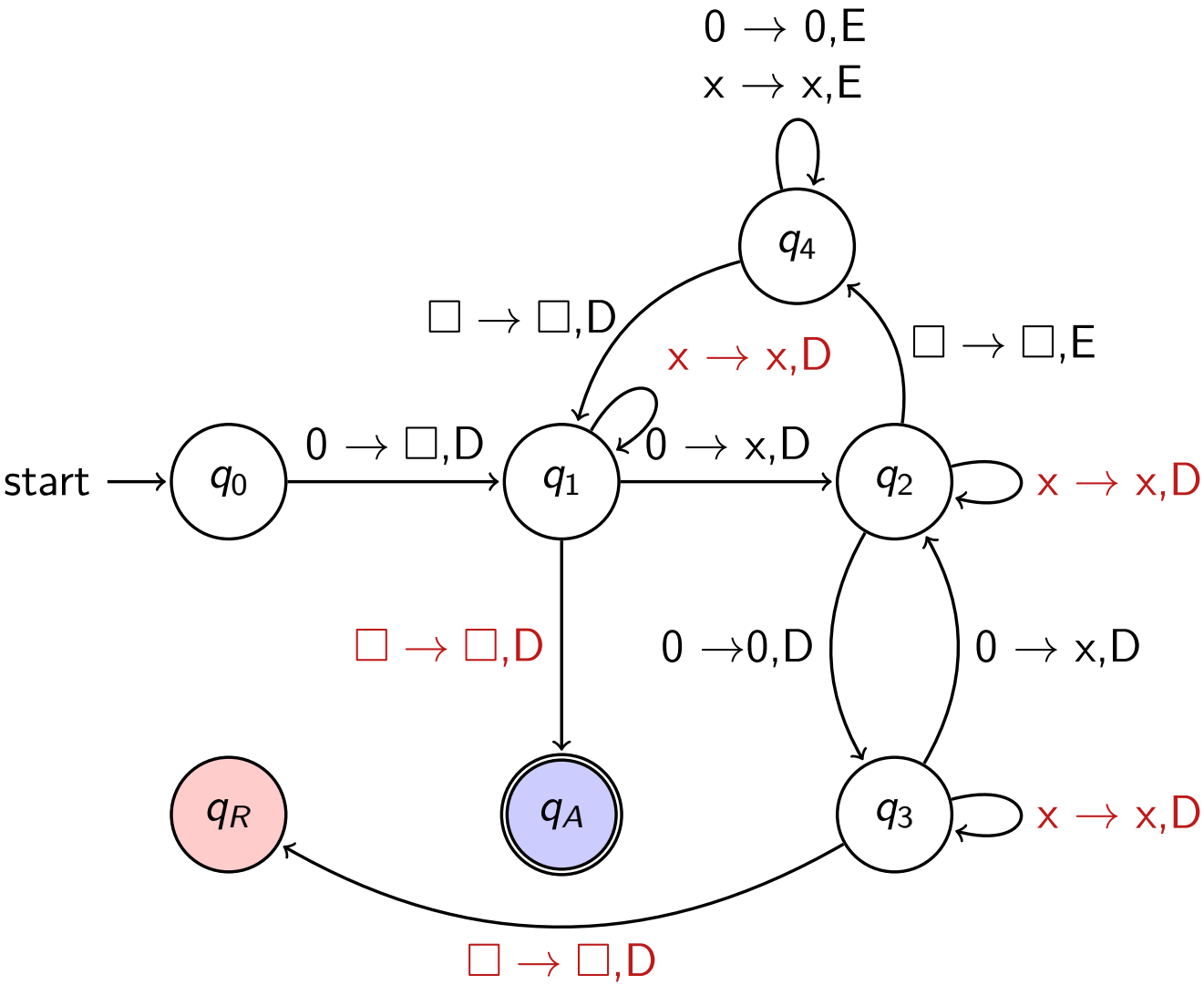


Continuando a entender a MT para $L_2 = \{0^{2^n} \mid n \geq 0\}$

Uma maneira ideal de entender esta MT é seguindo as transições do tratamento da cadeia 0000.
Até o estado grafado em vermelho corresponde a parte (1/2). Depois siga para (2/2).

q_0000
$\square q_1000$
$\square x q_200$
$\square x0q_30$
$\square x0xq_2\square$
$\square x0q_4x\square$
$\square xq_40x\square$
$\square q_4x0x\square$
$q_4\square x0x\square$
$\square q_1x0x\square$
$\square xq_10x\square$
$\square xxq_2x\square$
$\square xxxq_2\square$
$\square xxq_4x\square$
$\square xq_4xx\square$
$\square q_4xxx\square$
$q_4\square xxx\square$
$\square q_1xxx\square$
$\square xq_1xx\square$
$\square xxq_1x\square$
$\square xxxq_1\square$
$\square xxx\square q_A$

MT para reconhecer $L_2 = \{0^{2^n} \mid n \geq 0\}$ (2/2)



1 Recordação

2 Representação da MT

3 MT e linguagens

4 Variantes da MT
- MT multifitas
- MT não determinística

5 Exercício em sala

MT multifitas (1/2)

- Similar a uma MT ordinária mas contém várias fitas
- Cada fita tem sua própria cabeça de I/O
- Em tese, pode ocorrer um processamento paralelo para mais de uma entrada
- A função de transição permite a movimentação de todas as cabeças de I/O

$$\delta : Q \times \Gamma^k \rightarrow Q \times \Gamma^k \times \{E, D\}^k$$

- Para k o número de fitas

$$\delta(q_0, a_1, \ldots, a_k) = (q_i, b_1, \ldots, b_k, E, D, \ldots, E)$$

MT multifitas (2/2)

- Aparentemente são máquinas mais poderosas, mas
- Dois autômatos são equivalentes se reconhecem a mesma linguagem

Teorema

Para toda MT multifita existe uma MT equivalente de fita única.

1 Recordação

2 Representação da MT

3 MT e linguagens

4 Variantes da MT
- MT multifitas
- MT não determinística

5 Exercício em sala

MT não determinística

- Em qualquer ponto da computação a MT pode proceder de acordo com várias possibilidades
- A função de transição para a MT não determinística tem a forma

$$\delta : Q \times \Gamma \rightarrow P(Q \times \Gamma \times \{E, D\})$$

- A função computação para esta MT pode ser representada por uma árvore cujos ramos correspondem às diversas possibilidades
- Caso um ramo aceita leve ao estado de aceitação, a MT aceita a entrada

MT não determinística

Teorema

Para toda MT não determinística existe uma MT determinística que lhe é equivalente.

Noção de algoritmo

- Como vimos, desde Hilbert até 1936, a noção de algoritmo era apenas intuitiva
- Noção intuitiva era insuficiente para os avanços da Matemática
- Até aonde poderíamos calcular? Quais problemas são computáveis

Esquema da tese de Church-Turing

A noção intuitiva de algoritmos é igual ao conceito de algoritmos na Máquina de Turing.

1 Recordação

2 Representação da MT

3 MT e linguagens

4 Variantes da MT
- MT multifitas
- MT não determinística

5 Exercício em sala

Recordando...

Problema

Analisar uma MT M_3 para testar a pertinência da linguagem

$$L_3 = \{w\#w \mid w \in \{0,1\}^*\}$$

Condições:

- A entrada (palavra) pode ser muito longa
- Seja $s \in L_3$, $s = 011000\#011000$

Algoritmo para verificar $w\#w$

1. Faça uma varredura na cadeia de entrada para se assegurar que contenha somente um '#'. Caso contrário, rejeite
2. Faça um zigue-zague na fita para corresponder as posições nos dois lados de '#'. Marque-os a medida que correspondam
3. Depois de correspondidos todos os símbolos à esquerda de '#' verifique se ainda há algum símbolo sem correspondente. Caso haja, rejeite a cadeia, caso contrário aceite-a

Estratégia

Percorrer a fita movendo a cabeça para frente e para trás e analisar os símbolos das posições correspondentes.
Marcar os símbolos que são verificados

```
  |------------|
0 1 1 0 0 0 # 0 1 1 0 0 0
x 1 1 0 0 0 # x 1 1 0 0 0
    |------------|
x 1 1 0 0 0 # x 1 1 0 0 0
x x 1 0 0 0 # x x 1 0 0 0
      |------------|
x x 1 0 0 0 # x x 1 0 0 0
x x x 0 0 0 # x x x 0 0 0
```

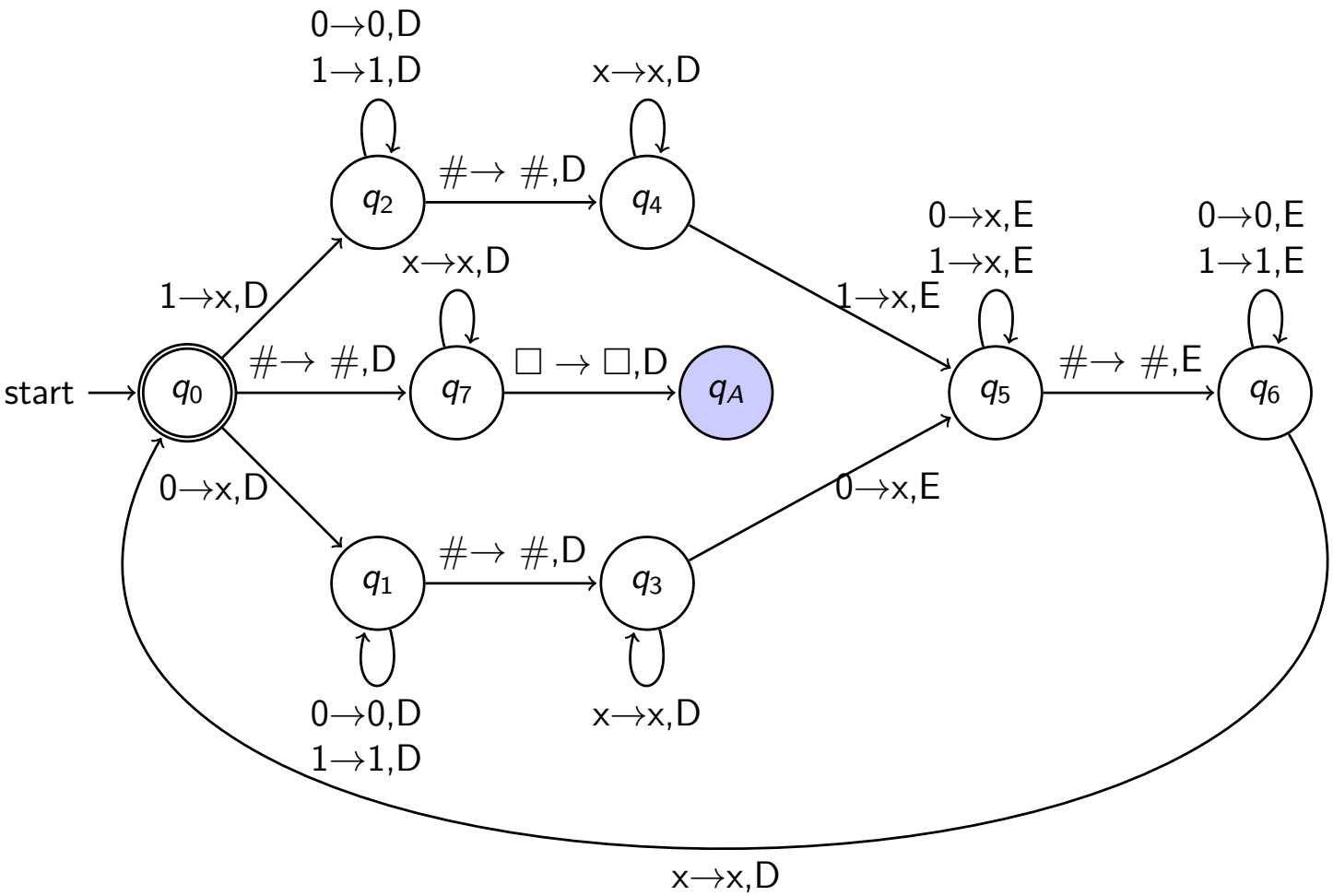
e assim segue. . .

Exercício

Já

Descreva uma MT que implemente o reconhecimento da linguagem $L_3 = \{w\#w \mid w \in \Sigma = \{0,1\}^*\}$

MT que reconhece $L_3 = \{w\#w \mid w \in \Sigma = \{0,1\}^*\}$



Finalizando. . .

Caros estudantes,

Leiam os livros texto e façam os exercícios sugeridos.