

ACH2043

INTRODUÇÃO À TEORIA DA COMPUTAÇÃO

Aula 12

Cap 3 – Máquinas de Turing

Profa. Arianne Machado Lima
arianne.machado@usp.br

Cap. 3

A tese de Church-Turing

Cap. 3 - A tese de Church-Turing

3.1 – Máquinas de Turing

3.2 – Variantes da Máquinas de Turing

3.3 – A Definição de Algoritmo

3.1 - Máquinas de Turing

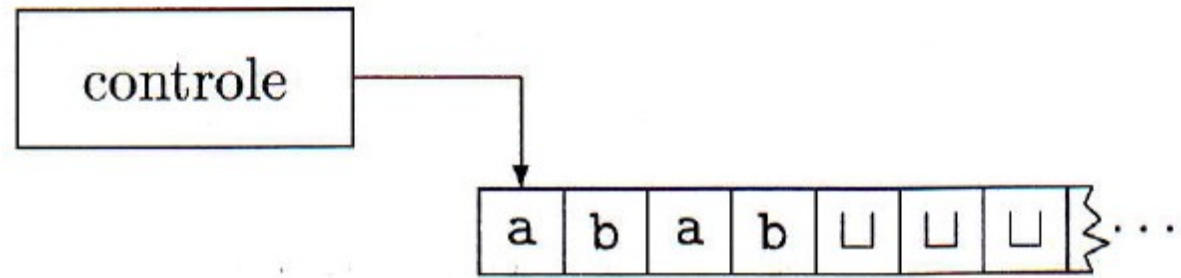
- Autômatos como modelos de computação:
 - AF: memória pequena
 - AP: memória ilimitada mas utilizável apenas em sistema LIFO (last in, first out) de leitura

3.1 - Máquinas de Turing

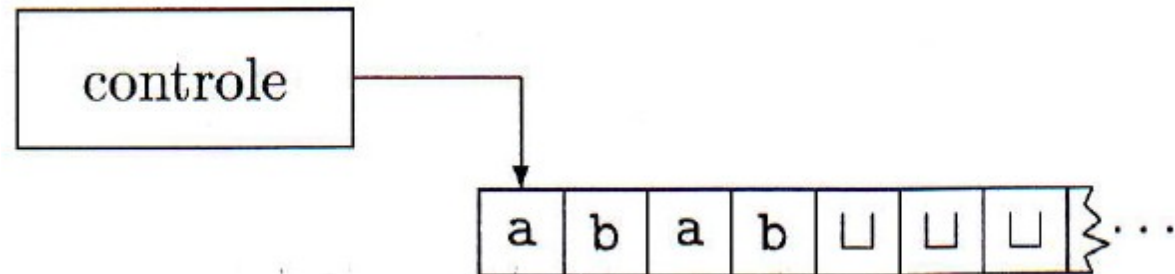
- Propostas por Alan Turing em 1936
 - Memória ilimitada e irrestrita
 - Modelo de um computador real (possibilidades e limitações)



Máquinas de Turing



Máquinas de Turing



1. Uma máquina de Turing pode tanto escrever sobre a fita quanto ler a partir dela.
2. A cabeça de leitura-escrita pode mover-se tanto para a esquerda quanto para a direita.
3. A fita é infinita.
4. Os estados especiais para rejeitar e aceitar fazem efeito imediatamente.

Máquinas de Turing - Exemplo

$B = \{ w \# w \mid w \text{ pertence a } \{0,1\}^* \}$

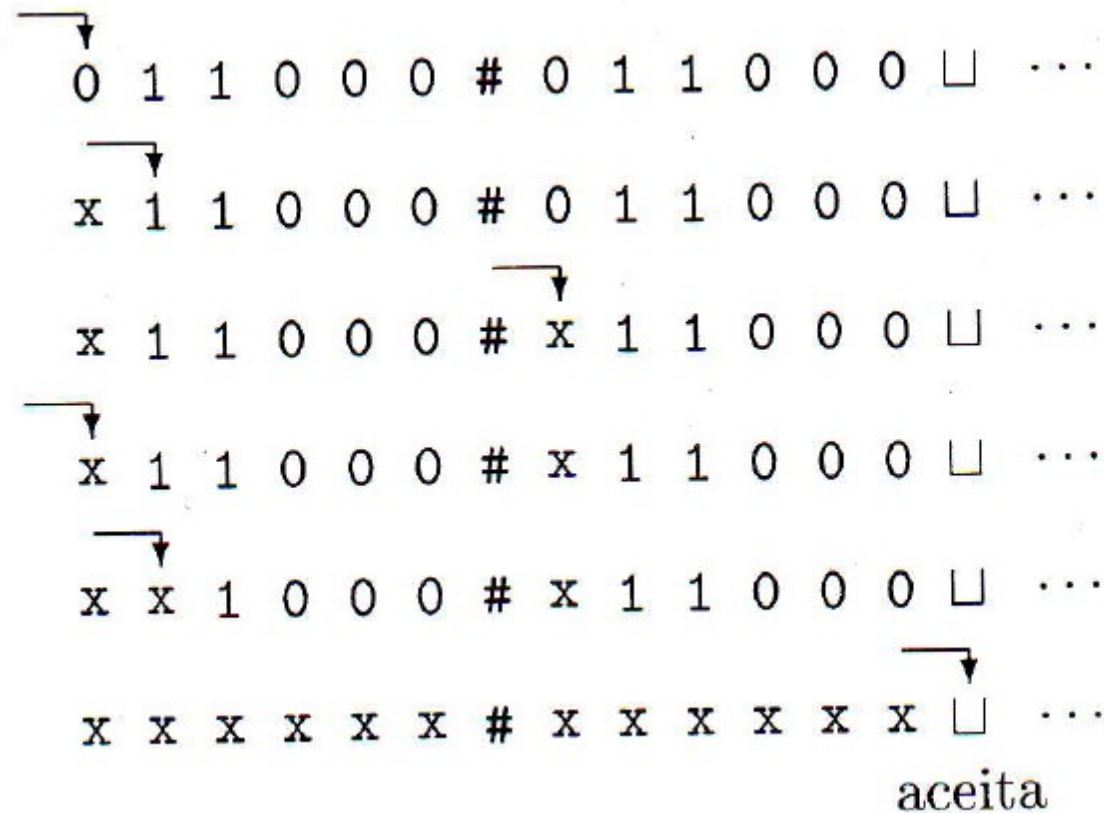
Máquinas de Turing - Exemplo

$$B = \{ w \# w \mid w \text{ pertence a } \{0,1\}^* \}$$

M_1 = “Sobre a cadeia de entrada w :

1. Faça um zigue-zague ao longo da fita checando posições correspondentes de ambos os lados do símbolo $\#$ para verificar se elas contêm o mesmo símbolo. Se elas não contêm, ou se nenhum $\#$ for encontrado, *rejeite*. Marque os símbolos à medida que eles são verificados para manter registro de quais símbolos têm correspondência.
2. Quando todos os símbolos à esquerda do $\#$ tiverem sido marcados, verifique a existência de algum símbolo remanescente à direita do $\#$. Se resta algum símbolo, *rejeite*; caso contrário, *aceite*.”

Máquinas de Turing - Exemplo



Máquinas de Turing – Definição formal

Uma *máquina de Turing* é uma 7-upla, $(Q, \Sigma, \Gamma, \delta, q_0, q_{\text{aceita}}, q_{\text{rejeita}})$, onde Q, Σ, Γ são todos conjuntos finitos e

1. Q é o conjunto de estados,
2. Σ é o alfabeto de entrada sem o *símbolo em branco* \sqcup ,
3. Γ é o alfabeto de fita, onde $\sqcup \in \Gamma$ e $\Sigma \subseteq \Gamma$,
4. $\delta: Q \times \Gamma \longrightarrow Q \times \Gamma \times \{E, D\}$ é a função de transição,
5. $q_0 \in Q$ é o estado inicial,
6. $q_{\text{aceita}} \in Q$ é o estado de aceitação, e
7. $q_{\text{rejeita}} \in Q$ é o estado de rejeição, onde $q_{\text{rejeita}} \neq q_{\text{aceita}}$.

Máquinas de Turing – Definição formal

Uma *máquina de Turing* é uma 7-upla, $(Q, \Sigma, \Gamma, \delta, q_0, q_{\text{aceita}}, q_{\text{rejeita}})$, onde Q, Σ, Γ são todos conjuntos finitos e

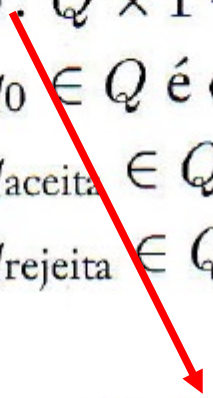
1. Q é o conjunto de estados,
2. Σ é o alfabeto de entrada sem o *símbolo em branco* \sqcup ,
3. Γ é o alfabeto de fita, onde $\sqcup \in \Gamma$ e $\Sigma \subseteq \Gamma$,
4. $\delta: Q \times \Gamma \longrightarrow Q \times \Gamma \times \{E, D\}$ é a função de transição,
5. $q_0 \in Q$ é o estado inicial,
6. $q_{\text{aceita}} \in Q$ é o estado de aceitação, e
7. $q_{\text{rejeita}} \in Q$ é o estado de rejeição, onde $q_{\text{rejeita}} \neq q_{\text{aceita}}$.

Cursor da fita vai para a esquerda ou direita

Máquinas de Turing – Definição formal

Uma *máquina de Turing* é uma 7-upla, $(Q, \Sigma, \Gamma, \delta, q_0, q_{\text{aceita}}, q_{\text{rejeita}})$, onde Q, Σ, Γ são todos conjuntos finitos e

1. Q é o conjunto de estados,
2. Σ é o alfabeto de entrada sem o *símbolo em branco* \sqcup ,
3. Γ é o alfabeto de fita, onde $\sqcup \in \Gamma$ e $\Sigma \subseteq \Gamma$,
4. $\delta: Q \times \Gamma \longrightarrow Q \times \Gamma \times \{E, D\}$ é a função de transição,
5. $q_0 \in Q$ é o estado inicial,
6. $q_{\text{aceita}} \in Q$ é o estado de aceitação, e
7. $q_{\text{rejeita}} \in Q$ é o estado de rejeição, onde $q_{\text{rejeita}} \neq q_{\text{aceita}}$.



$\delta: Q' \times \Gamma \longrightarrow Q \times \Gamma \times \{E, D\}$, onde Q' é Q sem q_{aceita} e q_{rejeita}

Máquinas de Turing - Funcionamento

- A entrada fica na porção mais à esquerda da fita
- O símbolo em branco marca o fim da entrada
- A máquina começa apontando para a primeira posição da fita
- Se a máquina está na primeira posição e tenta fazer um movimento para a esquerda, permanece no lugar
- Pára SOMENTE quando entra em um estado de aceitação ou rejeição

Máquinas de Turing - Funcionamento

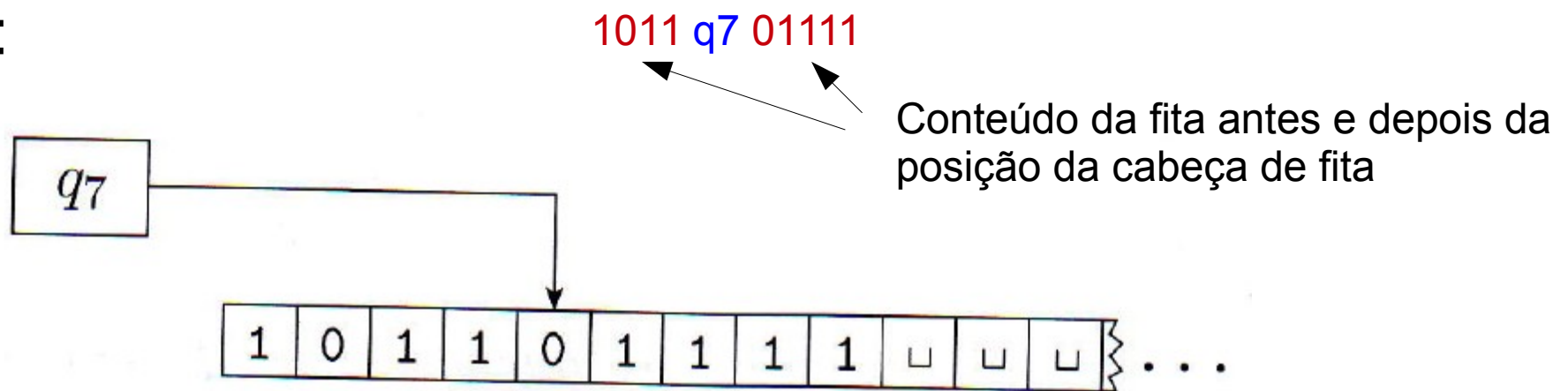
- **Configuração** - situação atual da máquina:

Máquinas de Turing - Funcionamento

- **Configuração** - situação atual da máquina:

- Estado atual
- Conteúdo da fita
- Posição da cabeça de fita

- Ex:



Máquinas de Turing - Funcionamento

Dizemos que uma configuração C_1 **origina** uma configuração C_2 se a máquina puder ir de C_1 a C_2 em um **único** passo.

Máquinas de Turing - Funcionamento

Suponha que tenhamos a, b e c em Γ , assim como u e v em Γ^* e os estados q_i e q_j . Nesse caso $uaq_i bv$ e $uq_j acv$ são duas configurações. Digamos que

$uaq_i bv$ origina $uq_j acv$

se na função de transição

Máquinas de Turing - Funcionamento

Suponha que tenhamos a, b e c em Γ , assim como u e v em Γ^* e os estados q_i e q_j . Nesse caso $uaq_i bv$ e $uq_j acv$ são duas configurações. Digamos que

$uaq_i bv$ origina $uq_j acv$

se na função de transição $\delta(q_i, b) = (q_j, c, E)$.

Máquinas de Turing - Funcionamento

Suponha que tenhamos a, b e c em Γ , assim como u e v em Γ^* e os estados q_i e q_j . Nesse caso $ua q_i bv$ e $u q_j acv$ são duas configurações. Digamos que

$ua q_i bv$ origina $u q_j acv$

se na função de transição $\delta(q_i, b) = (q_j, c, E)$.

$ua q_i bv$ origina $uac q_j v$

Máquinas de Turing - Funcionamento

Suponha que tenhamos a, b e c em Γ , assim como u e v em Γ^* e os estados q_i e q_j . Nesse caso $ua q_i bv$ e $u q_j acv$ são duas configurações. Digamos que

$$ua q_i bv \quad \text{origina} \quad u q_j acv$$

se na função de transição $\delta(q_i, b) = (q_j, c, E)$.

$$ua q_i bv \quad \text{origina} \quad uac q_j v$$

se $\delta(q_i, b) = (q_j, c, D)$.

Máquinas de Turing - Funcionamento

- Configuração inicial:
- Configuração de aceitação:
- Configuração de rejeição:

Máquinas de Turing - Funcionamento

- Configuração inicial: q_0w
- Configuração de aceitação:
- Configuração de rejeição:

Máquinas de Turing - Funcionamento

- Configuração inicial: q_0w
- Configuração de aceitação: estado atual = q_{aceita}
- Configuração de rejeição:

Máquinas de Turing - Funcionamento

- Configuração inicial: q_0w
- Configuração de aceitação: estado atual = q_{aceita}
- Configuração de rejeição: estado atual = $q_{rejeita}$

Máquinas de Turing - Funcionamento

- Configuração inicial: q_0w
- Configuração de aceitação: estado atual = q_{aceita}
- Configuração de rejeição: estado atual = q_{rejeita}

Uma máquina de
Turing M **aceita** a entrada w se uma seqüência de configurações C_1, C_2, \dots, C_k
existe, onde

Máquinas de Turing - Funcionamento

- Configuração inicial: q_0w
- Configuração de aceitação: estado atual = q_{aceita}
- Configuração de rejeição: estado atual = q_{rejeita}

Uma máquina de

Turing M **aceita** a entrada w se uma seqüência de configurações C_1, C_2, \dots, C_k existe, onde

1. C_1 é a configuração inicial de M sobre a entrada w ,
2. cada C_i origina C_{i+1} e
3. C_k é uma configuração de aceitação.

Máquinas de Turing

A coleção de cadeias que M aceita é *a linguagem de M* , ou *a linguagem reconhecida por M* , denotada $L(M)$.

DEFINIÇÃO 3.5

Chame uma linguagem de *Turing-reconhecível*, se alguma máquina de Turing a reconhece.¹

1 - Ou linguagem **recursivamente enumerável** ou linguagem **irrestrita**

Máquinas de Turing (MT) Decisoras

Uma MT é decisoras se ela nunca entra em loop (isto é, sempre pára em um estado de aceitação ou de rejeição).

Dizemos que um decisor que reconhece uma linguagem **decide** essa linguagem.

DEFINIÇÃO 3.6

Chame uma linguagem de *Turing-decidível* ou simplesmente *decidível* se alguma máquina de Turing a decide.²

2 - Ou linguagem **recursiva**

Máquinas de Turing - Exemplos

EXEMPLO 3.7

Aqui descrevemos uma máquina de Turing (MT) M_2 que decide $A = \{0^{2^n} \mid n \geq 0\}$, a linguagem consistindo em todas as cadeias de 0s cujo comprimento é uma potência de 2.

Ideia: Uma potência de 2, sempre que eu divido por 2, terei um número par

Máquinas de Turing - Exemplos

EXEMPLO 3.7

Aqui descrevemos uma máquina de Turing (MT) M_2 que decide $A = \{0^{2^n} \mid n \geq 0\}$, a linguagem consistindo em todas as cadeias de 0s cujo comprimento é uma potência de 2.

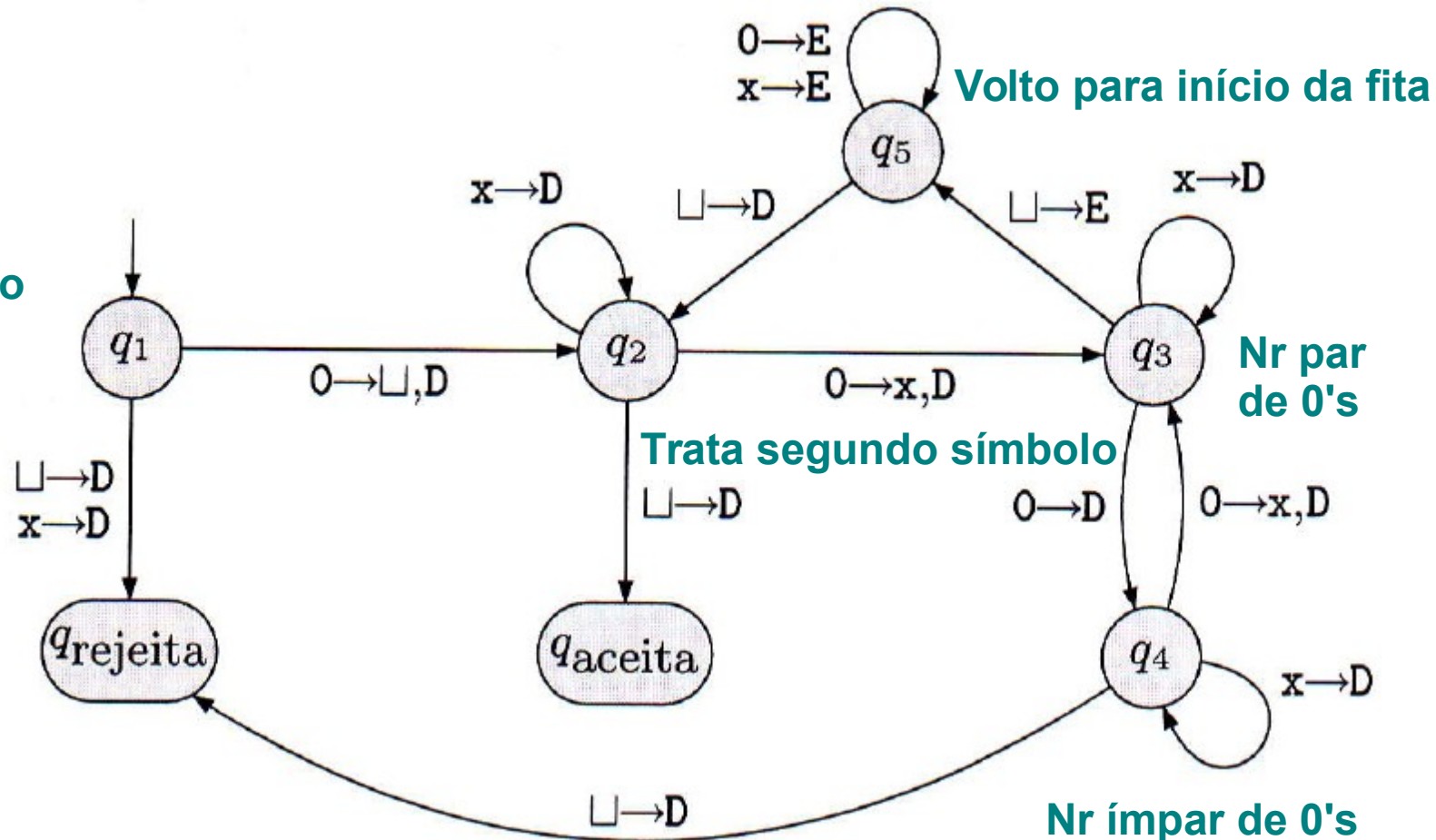
$M_2 =$ “Sobre a cadeia de entrada w :

1. Faça uma varredura da esquerda para a direita na fita, marcando um 0 não, e outro, sim.
2. Se no estágio 1, a fita continha um único 0, *aceite*.
3. Se no estágio 1, a fita continha mais que um único 0 e o número de 0s era ímpar, *rejeite*.
4. Retorne a cabeça para a extremidade esquerda da fita.
5. Vá para o estágio 1.”

Agora, damos a descrição formal de $M_2 = (Q, \Sigma, \Gamma, \delta, q_1, q_{aceita}, q_{rejeita})$:

- $Q = \{q_1, q_2, q_3, q_4, q_5, q_{aceita}, q_{rejeita}\}$,
- $\Sigma = \{0\}$ e
- $\Gamma = \{0, x, \sqcup\}$.
- Descrevemos δ com um diagrama de estados (veja a Figura 3.8).
- Os estados inicial, de aceitação e de rejeição são q_1 , q_{aceita} e $q_{rejeita}$.

Trata primeiro símbolo
na primeira iteração
(símbolo em branco
para indicar o início
da fita)



Exemplo para a cadeia 0000

$q_1 0000$

$\sqcup q_2 000$

$\sqcup x q_3 00$

$\sqcup x 0 q_4 0$

$\sqcup x 0 x q_3 \sqcup$

$\sqcup x 0 q_5 x \sqcup$

$\sqcup x q_5 0 x \sqcup$

$\sqcup q_5 x 0 x \sqcup$

$q_5 \sqcup x 0 x \sqcup$

$\sqcup q_2 x 0 x \sqcup$

$\sqcup x q_2 0 x \sqcup$

$\sqcup x x q_3 x \sqcup$

$\sqcup x x x q_3 \sqcup$

$\sqcup x x q_5 x \sqcup$

$\sqcup x q_5 x x \sqcup$

$\sqcup q_5 x x x \sqcup$

$q_5 \sqcup x x x \sqcup$

$\sqcup q_2 x x x \sqcup$

$\sqcup x q_2 x x \sqcup$

$\sqcup x x q_2 x \sqcup$

$\sqcup x x x q_2 \sqcup$

$\sqcup x x x \sqcup q_{aceita}$

Reverso esse exemplo

$$B = \{ w \# w \mid w \text{ pertence a } \{0,1\}^* \}$$

M_1 = “Sobre a cadeia de entrada w :

1. Faça um zigue-zague ao longo da fita checando posições correspondentes de ambos os lados do símbolo $\#$ para verificar se elas contêm o mesmo símbolo. Se elas não contêm, ou se nenhum $\#$ for encontrado, *rejeite*. Marque os símbolos à medida que eles são verificados para manter registro de quais símbolos têm correspondência.
2. Quando todos os símbolos à esquerda do $\#$ tiverem sido marcados, verifique a existência de algum símbolo remanescente à direita do $\#$. Se resta algum símbolo, *rejeite*; caso contrário, *aceite*.”

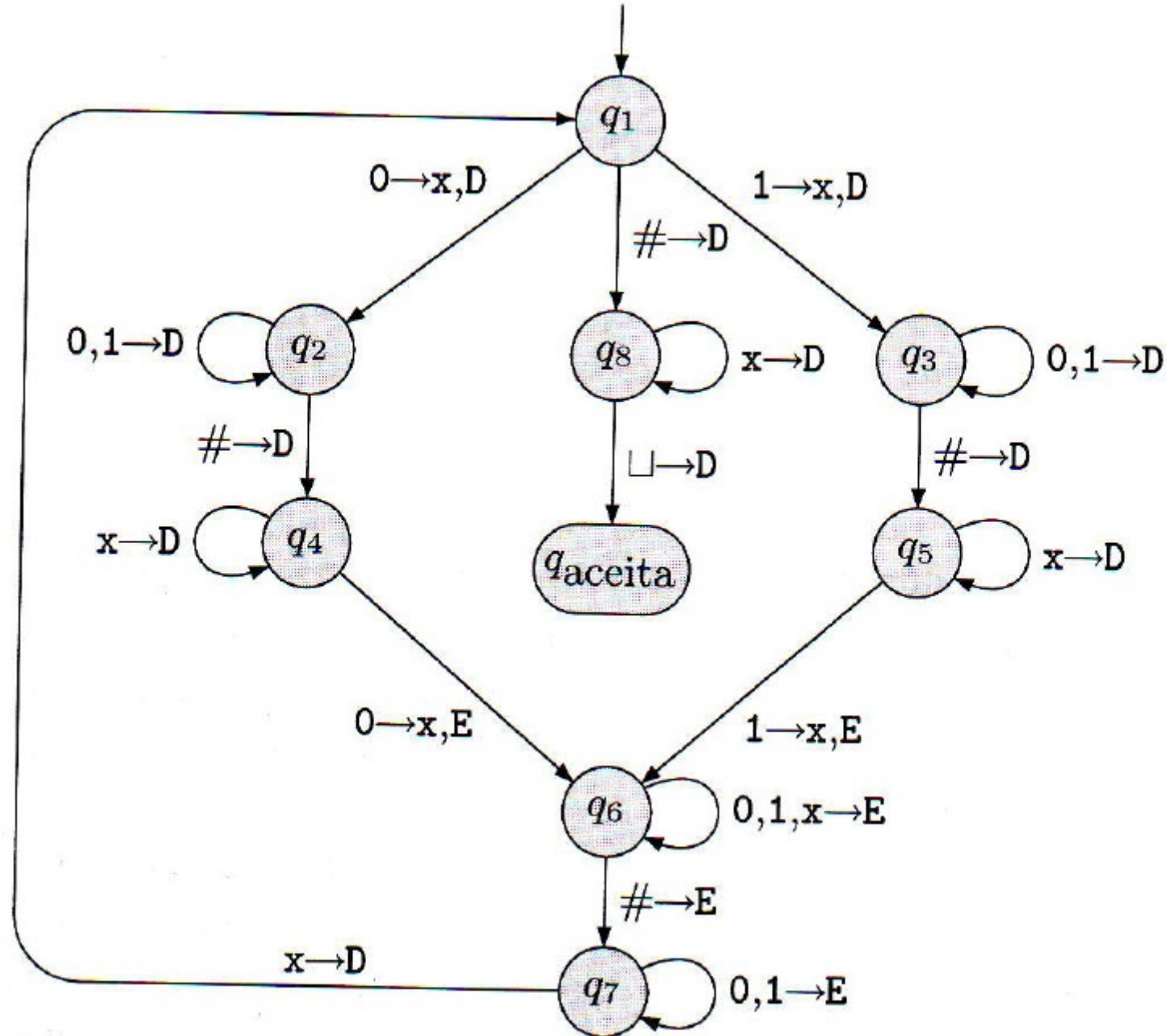
↙	0	1	1	0	0	0	#	0	1	1	0	0	0	□	...	
↙	x	1	1	0	0	0	#	0	1	1	0	0	0	□	...	
	x	1	1	0	0	0	#	↙	x	1	1	0	0	0	□	...
↙	x	1	1	0	0	0	#	x	1	1	0	0	0	□	...	
↙	x	x	1	0	0	0	#	x	1	1	0	0	0	□	...	
	x	x	x	x	x	x	#	x	x	x	x	x	x	↙	□	...

aceita

EXEMPLO 3.9

O que segue é uma descrição formal de $M_1 = (Q, \Sigma, \Gamma, \delta, q_1, q_{\text{aceita}}, q_{\text{rejeita}})$, a máquina de Turing que descrevemos informalmente na página 145, para decidir a linguagem $B = \{w\#w \mid w \in \{0,1\}^*\}$.

- $Q = \{q_1, \dots, q_{14}, q_{\text{aceita}}, q_{\text{rejeita}}\}$,
- $\Sigma = \{0,1,\#\}$, e $\Gamma = \{0,1,\#,x,\sqcup\}$.
- Descrevemos δ com um diagrama de estados (veja a figura seguinte).
- Os estados inicial, de aceitação e de rejeição são q_1 , q_{aceita} e q_{rejeita} .



Transições implícitas para $q_{rejeita}$ (indo para a direita, por convenção) quando aparece um símbolo não definido na transição.