INE5403

FUNDAMENTOS DE MATEMÁTICA DISCRETA PARA A COMPUTAÇÃO

PROF. DANIEL S. FREITAS

UFSC - CTC - INE

8.1) Linguagens

8.2) Máquinas de Estados Finitos

8.3) Monóides, Máquinas e Linguagens

LINGUAGENS & MÁQUINAS

- Determinar se uma string pertence à linguagem de uma dada gramática é, em geral, difícil
  - (em alguns casos é até impossível).
- Mas as gramáticas e linguagens regulares possuem propriedades que permitem construir um "reconhecedor" de strings.

LINGUAGENS & MÁQUINAS

- Uma máquina é um sistema que:
  - pode aceitar entradas ("input")
  - pode produzir saída ("output")
  - possui algum tipo de memória interna
    - capaz de "rastrear" informações sobre inputs anteriores

Linguagens & Máquinas

- Uma máquina é um sistema que:
  - pode aceitar entradas (“input”)
  - pode produzir saída (“output”)
  - possui algum tipo de memória interna
    - capaz de “rastrear” informações sobre inputs anteriores
- A qualquer momento em particular:

condição interna da máquina + sua memória	=	estado desta máquina naquele momento

LINGUAGENS & MÁQUINAS

- O estado de uma máquina a qualquer instante resume a sua memória de inputs passados
  - e determina como ela vai reagir a inputs subseqüentes

LINGUAGENS & MÁQUINAS

- O estado de uma máquina a qualquer instante resume a sua memória de inputs passados
  - e determina como ela vai reagir a inputs subseqüentes
- Quando chega mais input, o estado atual da máquina determina, junto com o próprio input:
  - o próximo estado a ser ocupado
  - qualquer output que possa ser produzido

LINGUAGENS & MÁQUINAS

- O estado de uma máquina a qualquer instante resume a sua memória de inputs passados
  - e determina como ela vai reagir a inputs subseqüentes
- Quando chega mais input, o estado atual da máquina determina, junto com o próprio input:
  - o próximo estado a ser ocupado
  - qualquer output que possa ser produzido
- Se o número destes estados é finito, esta é uma máquina de estados finitos.

MÁQUINAS DE ESTADOS FINITOS

- Sejam os conjuntos finitos $S = \{s_0, s_1, \ldots, s_n\}$ e I:
  - para cada $x \in I$, seja uma função $f_x : S \to S$
  - seja: $\mathcal{F} = \{f_x | x \in I\}$

MÁQUINAS DE ESTADOS FINITOS

- Sejam os conjuntos finitos $S = \{s_0, s_1, \ldots, s_n\}$ e I:
  - para cada $x \in I$, seja uma função $f_x : S \longrightarrow S$
  - seja: $\mathcal{F} = \{f_x | x \in I\}$
- A tripla $(S, I, \mathcal{F})$ é chamada de máquina de estados finitos:
  - S é o conjunto de estados
  - os elementos de S são chamados de estados
  - I é chamado de conjunto de entradas (inputs) da máquina

Máquinas de Estados Finitos

- Sejam os conjuntos finitos $S = \{s_0, s_1, \ldots, s_n\}$ e I:
  - para cada $x \in I$, seja uma função $f_x : S \longrightarrow S$
  - seja: $\mathcal{F} = \{f_x | x \in I\}$
- A tripla $(S, I, \mathcal{F})$ é chamada de máquina de estados finitos:
  - S é o conjunto de estados
  - os elementos de S são chamados de estados
  - I é chamado de conjunto de entradas (inputs) da máquina
  - para cada input $x \in I$:
    - f_x descreve o efeito deste input sobre os estados da máquina
    - é a chamada função de transição de estados

MÁQUINAS DE ESTADOS FINITOS

- Se a máquina está no estado s_i e o input x ocorre:
  - o próximo estado da máquina vai ser $f_x(s_i)$
- Uma vez que o próximo estado é determinado de maneira única pelo par (s_i, x):
  - existe uma função $F : S \times I \longrightarrow S$ dada por:

$$F(s_i, x) = f_x(s_i)$$

- Por esta razão:
  - alguns autores usam uma função $F : S \times I \longrightarrow S$ em vez de um conjunto $\{f_x | x \in I\}$ para definir uma M.E.F.
  - as definições são completamente equivalentes.

Máquinas de Estados Finitos

- **Exemplo 1:** sejam $S = \{s_0, s_1\}$ e $I = \{0, 1\}$.
  - Defina f_0 e f_1 como:

$$f_0(s_0) = s_0 \qquad f_1(s_0) = s_1$$

$$f_0(s_1) = s_1 \qquad f_1(s_1) = s_0$$

MÁQUINAS DE ESTADOS FINITOS

- **Exemplo 1:** sejam $S = \{s_0, s_1\}$ e $I = \{0, 1\}$.
  - Defina f_0 e f_1 como:

$$f_0(s_0) = s_0 \qquad f_1(s_0) = s_1$$

$$f_0(s_1) = s_1 \qquad f_1(s_1) = s_0$$

  - Esta M.E.F. possui dois estados (s_0 e s_1)
    - e aceita dois inputs (0 e 1)

MÁQUINAS DE ESTADOS FINITOS

- **Exemplo 1:** sejam $S = \{s_0, s_1\}$ e $I = \{0, 1\}$.
  - Defina f_0 e f_1 como:

$$f_0(s_0) = s_0 \qquad f_1(s_0) = s_1$$

$$f_0(s_1) = s_1 \qquad f_1(s_1) = s_0$$

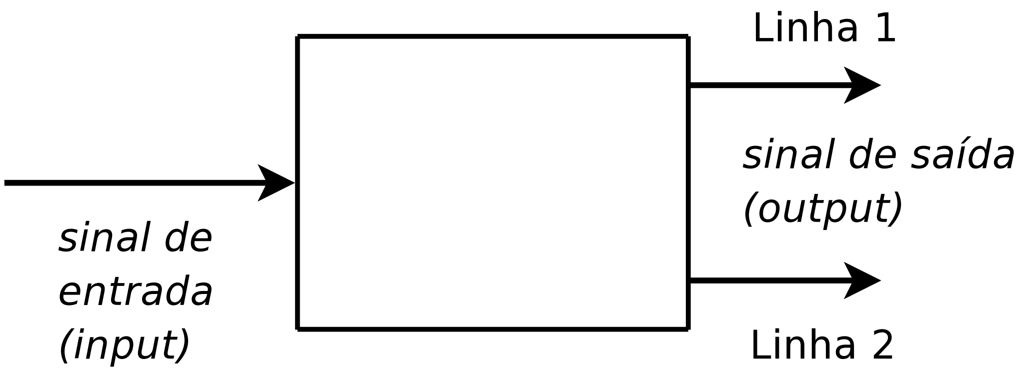
  - Esta M.E.F. possui dois estados (s_0 e s_1)
    - e aceita dois inputs (0 e 1):

      0: deixa os estados fixos

      1: reverte os estados

Máquinas de Estados Finitos

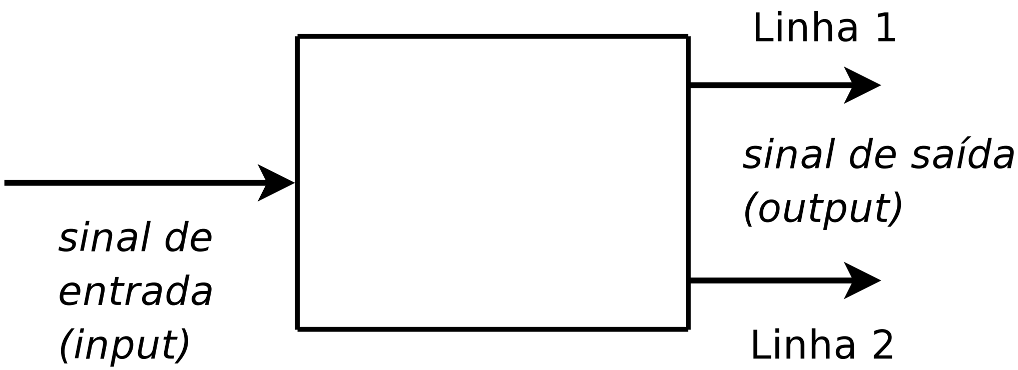
- A máquina do exemplo anterior pode servir como um modelo para o dispositivo lógico a seguir:



  - em um dado momento, sinais de saída consistem de duas voltagens, uma mais alta do que a outra
    - ou a linha 1 vai estar em voltagem mais alta e a 2 em mais baixa (s_0) ou o reverso (s_1)

MÁQUINAS DE ESTADOS FINITOS

- A máquina do exemplo 1 pode servir como um modelo para o dispositivo lógico a seguir:



- Um pulso na entrada (“1”) reverte as voltagens de saída.
- O símbolo (“0”) representa ausência de sinal na entrada e não resulta em modificação da saída
- Dispositivo frequentemente chamado de flip-flop
  - realização concreta da máquina do exemplo.

MÁQUINAS DE ESTADOS FINITOS

- Esta máquina pode ser resumida como:

	$\mathbf{0}$	$\mathbf{1}$
$\boldsymbol{s_0}$	$\boldsymbol{s_0}$	$\boldsymbol{s_1}$
$\boldsymbol{s_1}$	$\boldsymbol{s_1}$	$\boldsymbol{s_0}$

  - inputs estão listados no topo
  - coluna: função correspondente ao input

- Esta é a tabela de transição de estados da M.E.F.
  - modo conveniente de especificar a máquina.

MÁQUINAS DE ESTADOS FINITOS

- **Exemplo 2:** Seja a tabela de transição de estados:

	a	b
s_0	s_0	s_1
s_1	s_2	s_0
s_2	s_1	s_2

- Esta tabela mostra que:

$$f_a(s_0) = s_0, \quad f_a(s_1) = s_2, \quad f_a(s_2) = s_1$$

- e também que:

$$f_b(s_0) = s_1, \quad f_b(s_1) = s_0, \quad f_b(s_2) = s_2 \qquad \square$$

MÁQUINAS DE ESTADOS FINITOS

- Seja M uma M.E.F. com estados S, inputs I e funções de transição de estado $\{f_x | x \in I\}$.

- Relação (natural) R_M sobre S:
  - sejam $s_i, s_j \in S$
  - dizemos que $s_i \; R_M \; s_j$ se há um input x tal que:
  $$f_x(s_i) = s_j$$
  - $s_i \; R_M \; s_j$ significa que, se a máquina está no estado s_i, $\exists x \in I$ tal que, se recebido em seguida, coloca a máq. no estado s_j

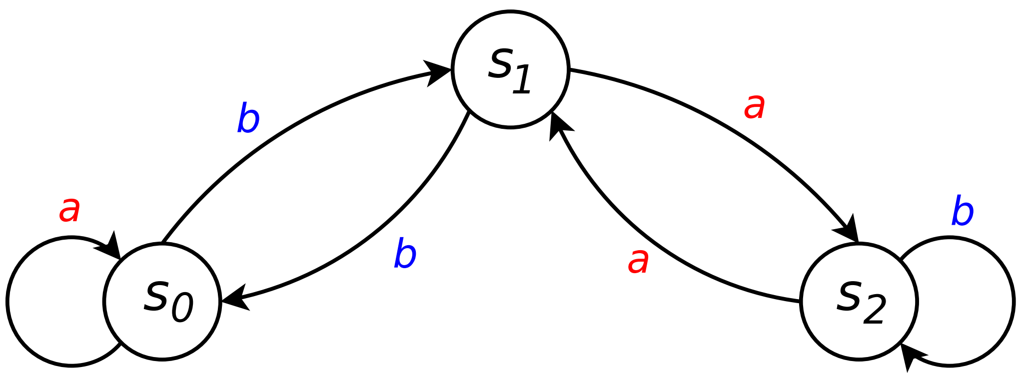
MÁQUINAS DE ESTADOS FINITOS

- Seja M uma M.E.F. com estados S, inputs I e funções de transição de estado $\{f_x | x \in I\}$.
- Relação (natural) R_M sobre S:
  - sejam $s_i, s_j \in S$
  - dizemos que $s_i \; R_M \; s_j$ se há um input x tal que:
    $$f_x(s_i) = s_j$$
  - $s_i \; R_M \; s_j$ significa que, se a máquina está no estado s_i, $\exists x \in I$ tal que, se recebido em seguida, coloca a máq. no estado s_j
- R_M permite descrever M como um dígrafo:
  - cada aresta é rotulada pelo conjunto de todos os inputs que fazem com que a máquina mude estados como indicado.

MÁQUINAS DE ESTADOS FINITOS

- **Exemplo 3:** Dígrafo da relação R_M para a máquina do exemplo 2:

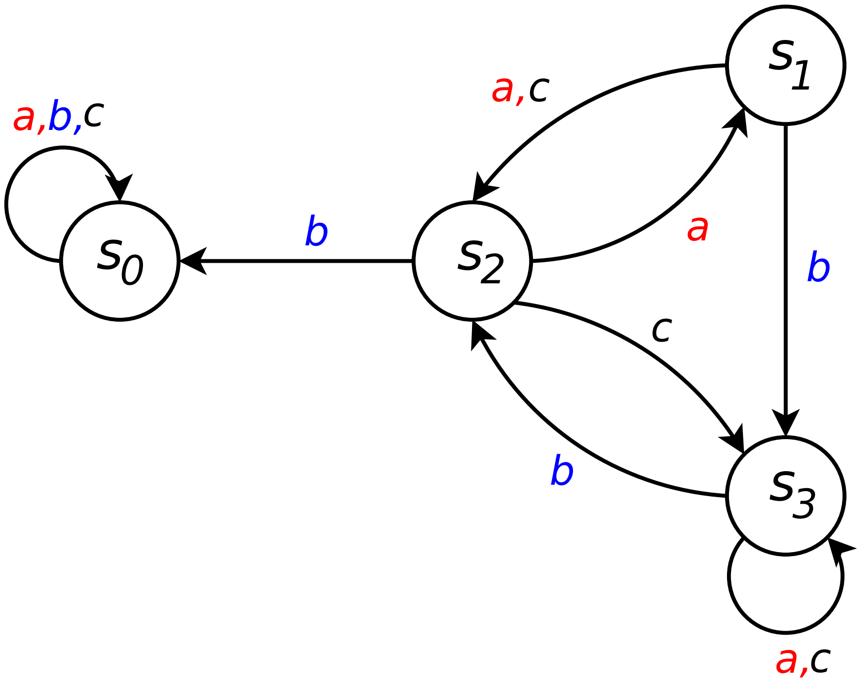
	a	b
s_0	s_0	s_1
s_1	s_2	s_0
s_2	s_1	s_2



MÁQUINAS DE ESTADOS FINITOS

- **Exemplo 4:** Dígrafo da relação R_M para a máquina M da tabela dada:

	a	b	c
s_0	s_0	s_0	s_0
s_1	s_2	s_3	s_2
s_2	s_1	s_0	s_3
s_3	s_3	s_2	s_3



MÁQUINAS DE ESTADOS FINITOS

- Note que uma aresta pode ser rotulada por mais do que um input.
  - Vários inputs podem causar a mesma mudança de estado.
- Além disto: todo input deve rotular exatamente uma aresta saindo de cada estado.

MÁQUINAS DE ESTADOS FINITOS

- Pode-se adicionar diversos recursos extras a uma M.E.F. para aumentar a utilidade do conceito.
- Uma extensão simples e muito útil leva à chamada Máquina de Moore (ou “máquina reconhecedora”).

MÁQUINAS DE ESTADOS FINITOS

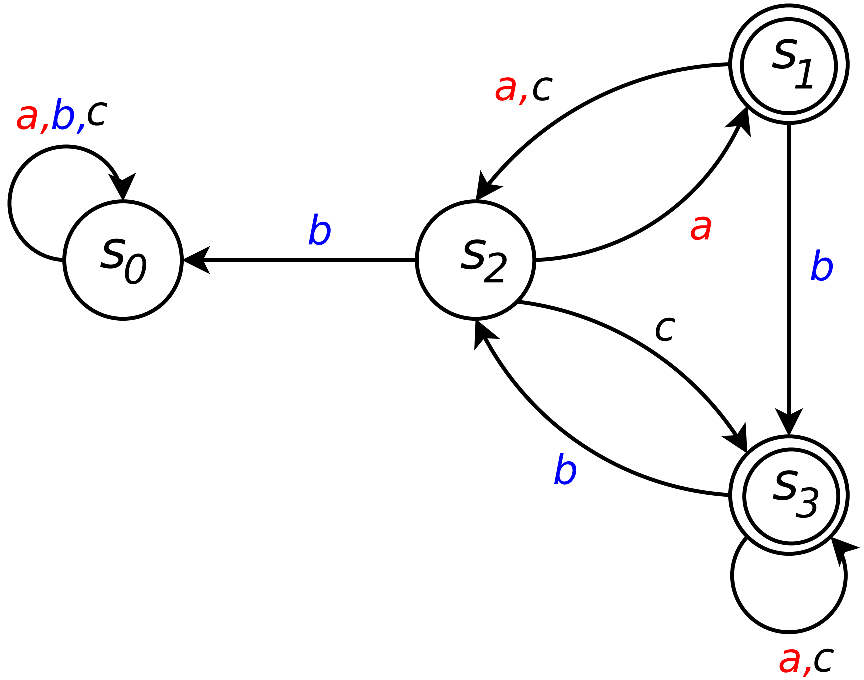
- Pode-se adicionar diversos recursos extras a uma M.E.F. para aumentar a utilidade do conceito.
- Uma extensão simples e muito útil leva à chamada Máquina de Moore (ou "máquina reconhecedora").
- A máquina de Moore é definida como $(S, I, \mathcal{F}, s_0, T)$, aonde:
  - $(S, I, \mathcal{F})$ é uma Máquina de Estados Finitos.
  - $s_0 \in S$: é o estado de partida de M
    - condição da máquina antes de receber qualquer input
  - $T \subseteq S$: conjunto de estados de aceitação de M
    - conexão com reconhecimento de linguagens

MÁQUINAS DE MOORE

- No dígrafo de uma máquina de Moore, os estados de aceitação têm dois círculos concêntricos.
- O estado de partida não recebe notação especial
  - mas normalmente será o s_0.

MÁQUINAS DE MOORE

- **Exemplo 5:** Dígrafo da máq. de Moore $(S, I, \mathcal{F}, s_0, T)$, aonde $(S, I, \mathcal{F})$ é a máq. do exemplo 4 e $T = \{s_1, s_3\}$:



MÁQUINAS QUOCIENTE

- Seja $M = (S, I, \mathcal{F})$ uma M.E.F. e R uma relação de equivalência sobre S:
  - dizemos que R é uma congruência de máquina sobre M se:

$$\forall s, t \in S, \quad s \, R \, t \ \Rightarrow \ f_x(s) \, R \, f_x(t), \quad \forall x \in I$$

  - "pares equivalentes de estados são levados para pares equivalentes de estados por todo input em I"

MÁQUINAS QUOCIENTE

- Se R é uma congruência de máquina sobre $M = (S, I, \mathcal{F})$:
  - definimos $\overline{S} = S/R$ como a partição de S correspondente a R
  - então: $\overline{S} = \{[s] \mid s \in S\}$

MÁQUINAS QUOCIENTE

- Se R é uma congruência de máquina sobre $M = (S, I, \mathcal{F})$:
  - definimos $\overline{S} = S/R$ como a partição de S correspondente a R
  - então: $\overline{S} = \{[s] \mid s \in S\}$

- Para todo input $x \in I$, definimos a relação $\overline{f_x}$ sobre $\overline{S}$:

$$\overline{f_x} = \{(\ [s]\ ,\ [f_x(s)]\ )\}$$

  - $[s] = [t] \ \Rightarrow\ s\ R\ t \ \Rightarrow\ f_x(s)\ R\ f_x(t) \ \Rightarrow\ [f_x(s)] = [f_x(t)]$
  - de modo que a relação $\overline{f_x}$ é uma função de $\overline{S}$ para S
    - e podemos escrever: $\overline{f_x}([s]) = [f_x(s)]$

MÁQUINAS QUOCIENTE

- A tripla $\overline{M} = (\overline{S}, I, \overline{\mathcal{F}})$, aonde $\overline{\mathcal{F}} = \{\overline{f_x} \mid x \in I\}$, é uma M.E.F..
  - é o chamado quociente de M correspondente a R
  - denotamos $\overline{M}$ por M/R
- Em geral, as máquinas quociente são mais simples do que as originais.
  - é frequentemente possível encontrar uma máq. quociente mais simples que substitui a original para alguns propósitos.

MÁQUINAS QUOCIENTE

- **Exemplo 6:** Seja M uma M.E.F. com tabela de transição de estados sobre $S = \{s_0, s_1, s_2, s_3, s_4, s_5\}$ dada por:

	a	b
s_0	s_0	s_4
s_1	s_1	s_0
s_2	s_2	s_4
s_3	s_5	s_2
s_4	s_4	s_3
s_5	s_3	s_2

MÁQUINAS QUOCIENTE

- **Exemplo 6:** Seja M uma M.E.F. sobre $S = \{s_0, s_1, s_2, s_3, s_4, s_5\}$:
  - e seja a R a relação de equivalência sobre S cuja matriz é:

$$M_R = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Máquinas Quociente

- **Exemplo 6:** Seja M uma M.E.F. sobre $S = \{s_0, s_1, s_2, s_3, s_4, s_5\}$:
  - então temos $S/R = \{[s_0], [s_1], [s_4]\}$, aonde:

    $[s_0] = \{s_0, s_2\} = [s_2]$

    $[s_1] = \{s_1, s_3, s_5\} = [s_3] = [s_5]$

    $[s_4] = \{s_4\}$

MÁQUINAS QUOCIENTE

- **Exemplo 6:** Seja M uma M.E.F. sobre $S = \{s_0, s_1, s_2, s_3, s_4, s_5\}$:
  - então temos $S/R = \{[s_0], [s_1], [s_4]\}$, aonde:

    $[s_0] = \{s_0, s_2\} = [s_2]$

    $[s_1] = \{s_1, s_3, s_5\} = [s_3] = [s_5]$

    $[s_4] = \{s_4\}$

  - a tabela de transição de estados mostra que f_a leva cada elemento de $[s_i]$ para outro elemento de $[s_i]$, para $i = 0, 1, 4$

Máquinas Quociente

- **Exemplo 6:** Seja M uma M.E.F. sobre $S = \{s_0, s_1, s_2, s_3, s_4, s_5\}$:
  - então temos $S/R = \{[s_0], [s_1], [s_4]\}$, aonde:

    $$[s_0] = \{s_0, s_2\} = [s_2]$$

    $$[s_1] = \{s_1, s_3, s_5\} = [s_3] = [s_5]$$

    $$[s_4] = \{s_4\}$$

  - a tabela de transição de estados mostra que f_a leva cada elemento de $[s_i]$ para outro elemento de $[s_i]$, para $i = 0, 1, 4$
  - além disto, f_b leva:
    - cada elemento de $[s_0]$ para um elemento de $[s_4]$
    - e cada elemento de $[s_4]$ para um elemento de $[s_1]$

MÁQUINAS QUOCIENTE

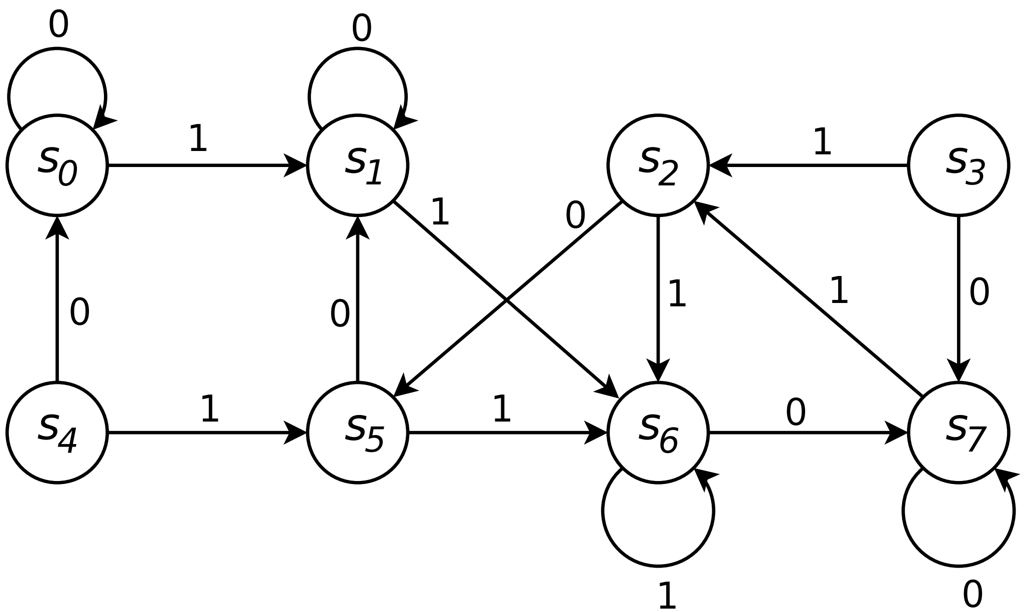
- **Exemplo 6:** Seja M uma M.E.F. sobre $S = \{s_0, s_1, s_2, s_3, s_4, s_5\}$:
  - logo, R é uma congruência de máquina
  - e a tabela de transição de estados de M/R é dada por:

	a	b
$[s_0]$	$[s_0]$	$[s_4]$
$[s_1]$	$[s_1]$	$[s_0]$
$[s_4]$	$[s_4]$	$[s_2]$

$\square$

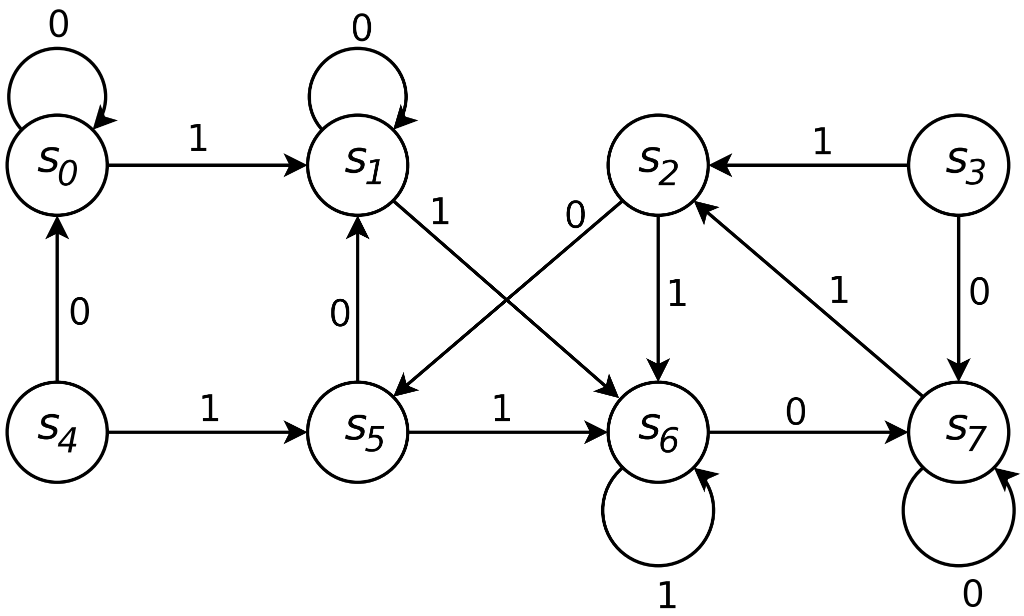
MÁQUINAS QUOCIENTE

- **Exemplo 7:** Seja $I = \{0, 1\}$, $S = \{s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7\}$, e $M = (S, I, \mathcal{F})$ a M.E.F. cujo dígrafo é:



MÁQUINAS QUOCIENTE

- **Exemplo 7:** Seja $I = \{0, 1\}$, $S = \{s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7\}$, e $M = (S, I, \mathcal{F})$ a M.E.F. cujo dígrafo é:



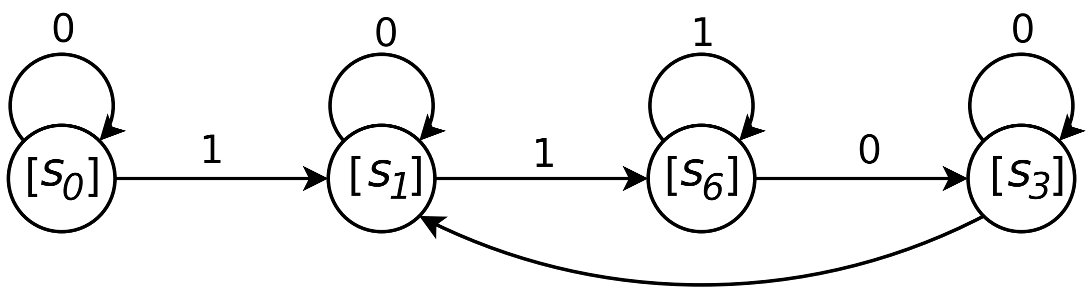
- Seja R tal que $S/R = \{\{s_0, s_4\}, \{s_1, s_2, s_5\}, \{s_6\}, \{s_3, s_7\}\}$
- o dígrafo de M mostra que R é uma congruência de máquina

MÁQUINAS QUOCIENTE

- **Exemplo 7 (cont.):** $I = \{0, 1\}$, $S = \{s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7\}$.
  - para obter o dígrafo da máquina quociente $\overline{M}$:
    - desenhar um vértice para cada classe de equivalência
    - ligar $[s_i]$ com $[s_j]$ se existir, no grafo original, uma aresta de algum vértice em $[s_i]$ para algum vértice em $[s_j]$

Máquinas Quociente

- **Exemplo 7 (cont.):** $I = \{0, 1\}$, $S = \{s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7\}$.
  - para obter o dígrafo da máquina quociente $\overline{M}$:
    - desenhar um vértice para cada classe de equivalência
    - ligar $[s_i]$ com $[s_j]$ se existir, no grafo original, uma aresta de algum vértice em $[s_i]$ para algum vértice em $[s_j]$
    - o resultado é:



□

MÁQUINAS DE MOORE QUOCIENTE

- Seja $M = (S, I, \mathcal{F}, s_0, T)$ uma Máquina de Moore.
- E seja R uma congruência de máquina sobre M.
- Então podemos fazer: $\overline{T} = \{[t] \mid t \in T\}$
- E a seqüência $\overline{M} = (\overline{S}, I, \overline{\mathcal{F}}, [s_0], \overline{T})$ é uma máquina de Moore.

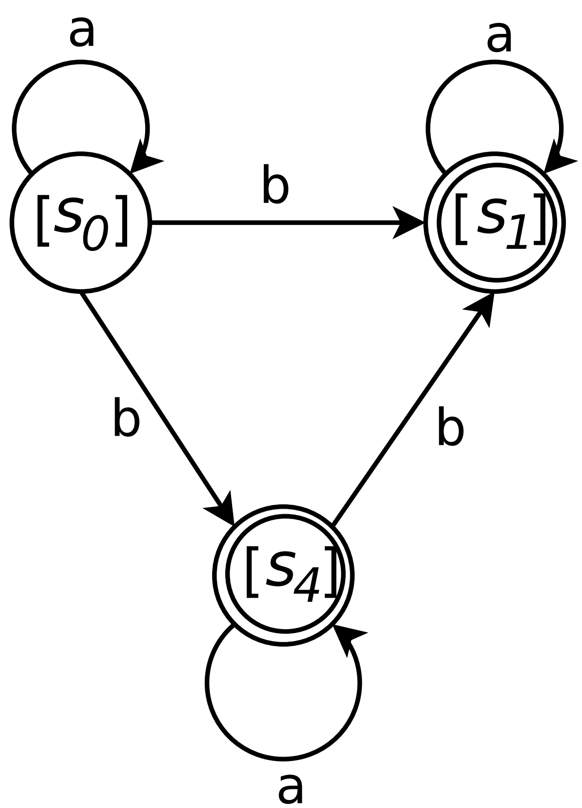
MÁQUINAS DE MOORE QUOCIENTE

- Seja $M = (S, I, \mathcal{F}, s_0, T)$ uma Máquina de Moore.
- E seja R uma congruência de máquina sobre M.
- Então podemos fazer: $\overline{T} = \{[t] \mid t \in T\}$
- E a seqüência $\overline{M} = (\overline{S}, I, \overline{\mathcal{F}}, [s_0], \overline{T})$ é uma máquina de Moore.
- Ou seja:
  - computamos a máquina quociente usual M/R
  - designamos $[s_0]$ como um estado de partida
  - daí, fazemos $\overline{T}$ o conjunto das classes de equivalência dos estados de aceitação
  - a máquina de Moore resultante $\overline{M}$ é a Máquina de Moore quociente de M.

MÁQUINAS QUOCIENTE

- **Exemplo 8:** Considere a máquina de Moore $M = (S, I, \mathcal{F}, s_0, T)$, aonde $M = (S, I, \mathcal{F})$ é a M.E.F. do exemplo 6 e $T = \{s_1, s_3, s_4\}$

  O dígrafo da máquina de Moore resultante é:



Máquinas de Estados Finitos

- Final deste item.
- **Dica**: fazer exercícios...