Linguagens Formais e Autômatos

Prof. Alex Luciano Roesler Rese, MSc.
Adaptado: Rafael de Santiago, Dr.

Linguagens Regulares

Linguagens Regulares

- Linguagens mais simples (hierarquia de Chomsky), sendo possível desenvolver algoritmos de reconhecimento, de geração ou de conversão entre formalismos de pouca complexidade, de grande eficiência e de fácil implementação

Linguagens Regulares

- Nem toda uma linguagem é uma linguagem regular!

- Exemplo dos parênteses

Exemplo:
(2 + 5) * 7
Representam ordem de Prioridades

Linguagens Regulares

- Sistemas de estados finitos: modelo matemático de sistema com entradas e saídas discretas. Podem assumir um número de estados finitos, necessários para determinar ações da próxima entrada

UNIVALI

Linguagens Regulares

- Composição de um sistema:
  - Seqüêncial: a execução da próxima componente depende da terminação da componente anterior
  - Concorrente: resulta componentes independentes, no qual a ordem de execução não é importante
  - Não-determinista: a próxima componente a ser executada é uma escolha entre diversas componentes alternativas

Autômato Finito

Autômato Finito

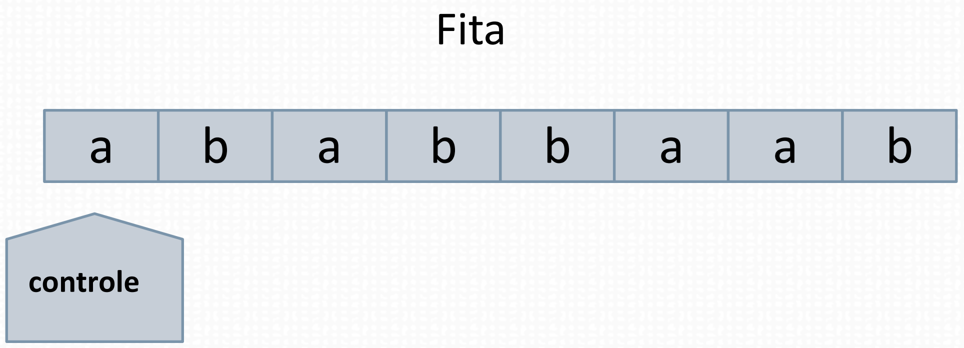
- Sistema de estados

- Seqüêncial

Autômato Finito

- Mecanismo formal composto de:
  - **Fita**: Dispositivo de entrada que tem a informação a palavra a ser reconhecida;
  - **Unidade de Controle**: Reflete o estado corrente da máquina. possui um cabeçote de leitura da fita e move-se sempre para direita
  - **Função de transição**: Função que comanda as leituras e define o estado da máquina.

Autômato Finito



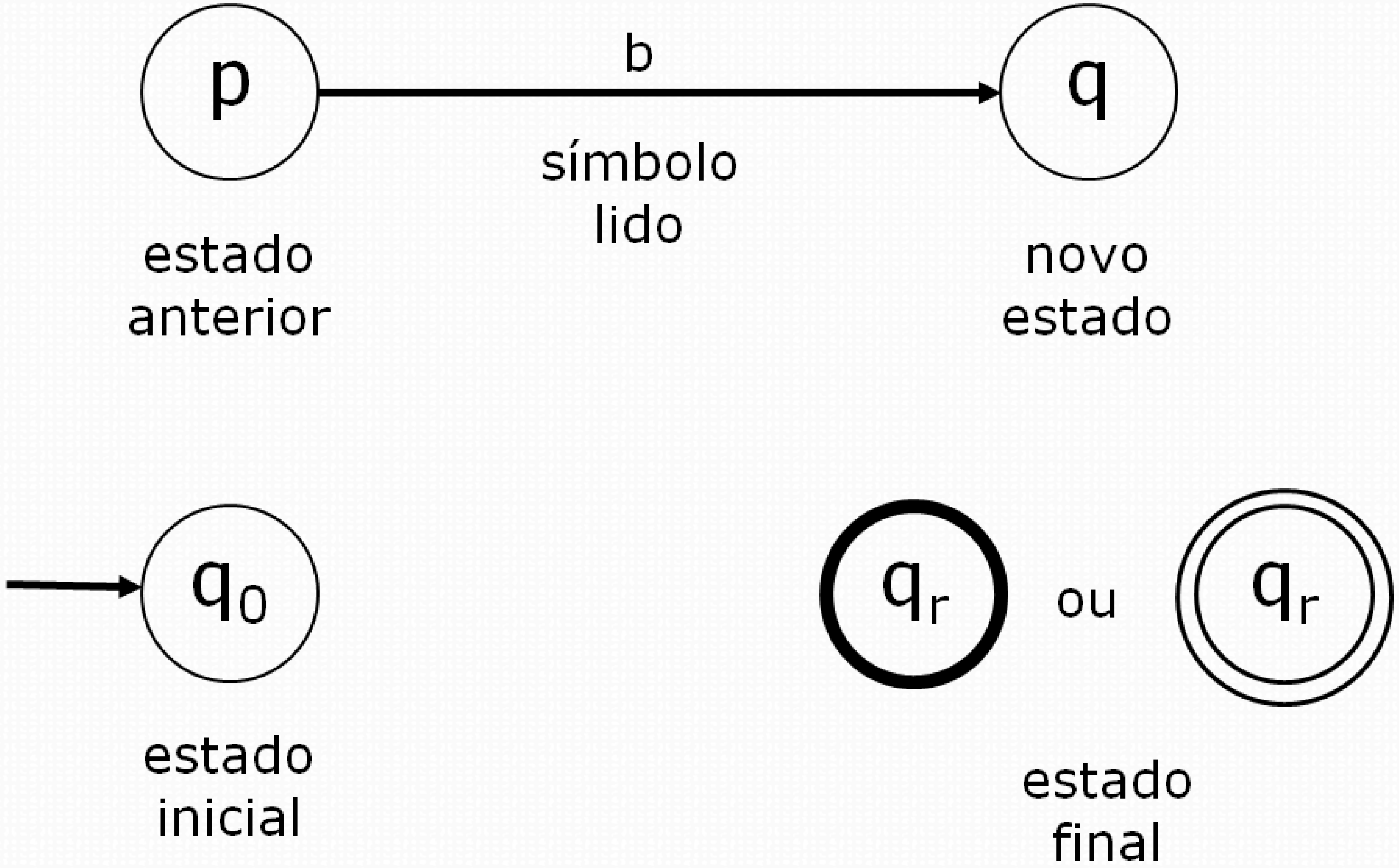
- A unidade de controle lê um símbolo e move-se para a célula da direita.
- O estado corrente pode mudar conforme definido pela função de transição e depende do estado atual e do símbolo lido.

Autômato Finito

- Um AFD é definidos por uma 5-upla:
- $M = (\Sigma, Q, \delta, q_0, F)$
  - Σ = alfabeto de símbolos de entrada
  - Q = conjunto finito de estados possíveis do autômato
  - δ = função de transição tal que $\delta: Q \times \Sigma \rightarrow Q$
  - q_0 = estado inicial tal que $q_0 \in Q$
  - F = conjunto de estados finais tal que $F \subset Q$

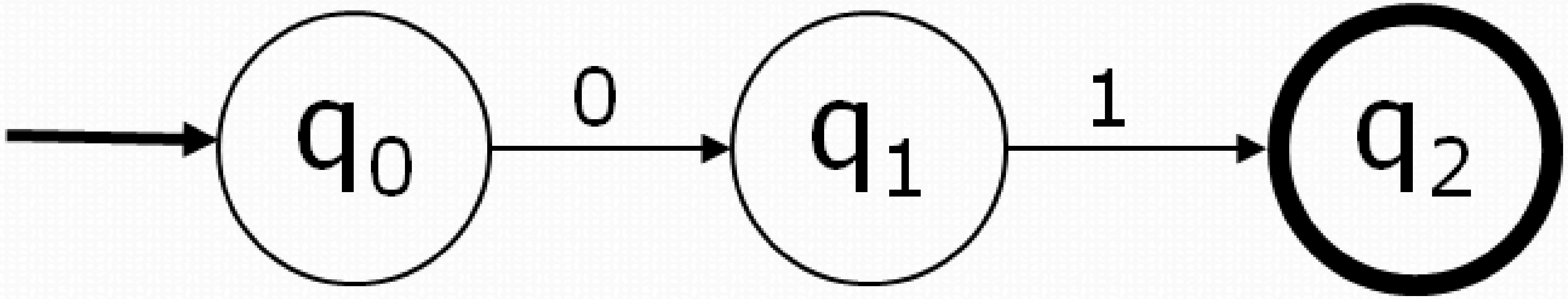
Grafo para Função de Transição

- Pode ser representado por um grafo onde:



Exemplos de Linguagens Regulares

- $M = (\Sigma, Q, \delta, q_0, F)$
- $M = (\{0,1\}, \{q_0,q_1,q_2\}, \delta, q_0, q_2)$

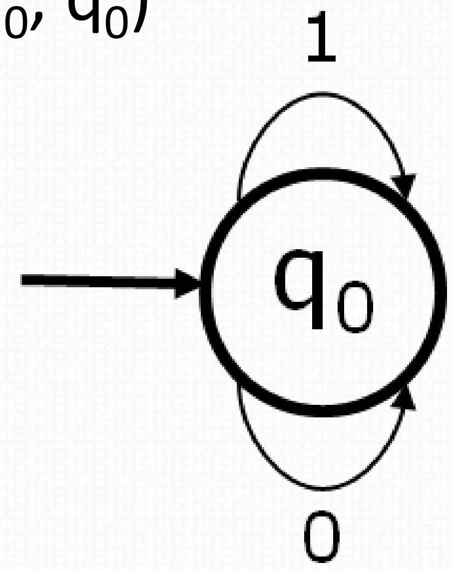


Linguagem que aceita apenas a palavra "01"

Exemplo de Linguagens Regulares

$M = (\Sigma, Q, \delta, q_0, F)$

$M = (\{0,1\}, \{q_0\}, \delta, q_0, q_0)$

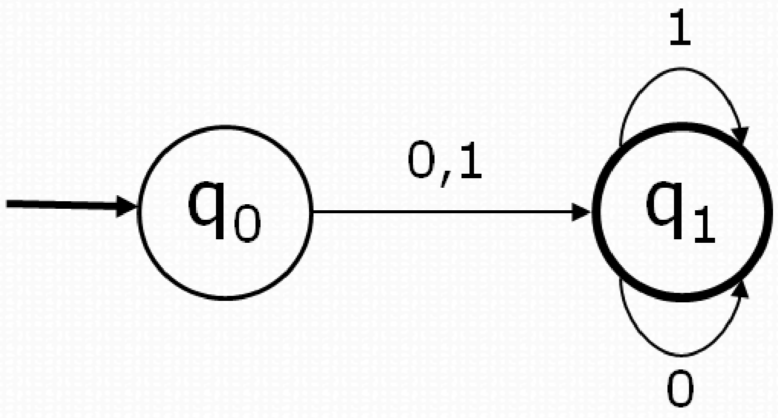


Linguagem que aceita ε e qualquer número binário

Exemplo de Linguagens Regulares

$M = (\Sigma, Q, \delta, q_0, F)$

$M = (\{0,1\}, \{q_0,q_1\}, \delta, q_0, q_1)$

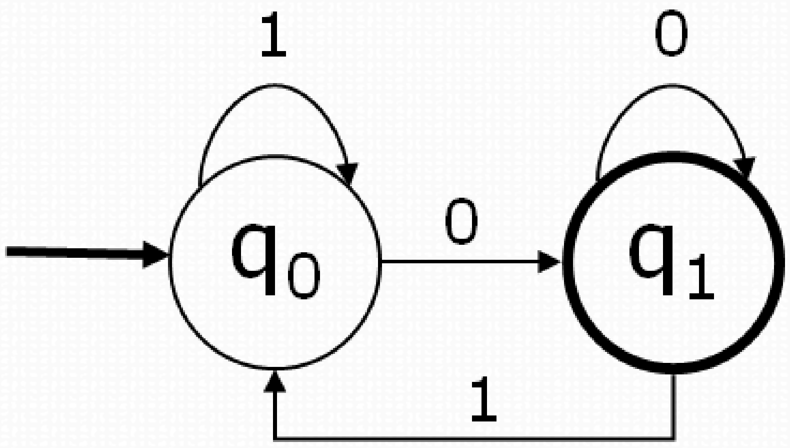


Linguagem que aceita números binários

Exemplos de Linguagens Regulares

$M = (\Sigma, Q, \delta, q_0, F)$

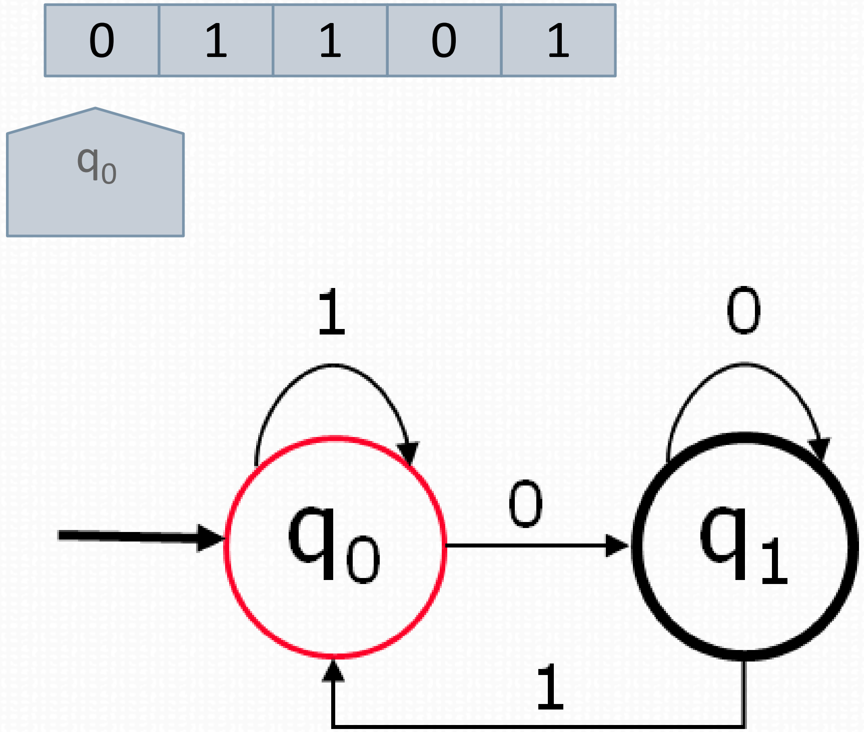
$M = (\{0,1\}, \{q_0,q_1\}, \delta, q_0, q_1)$



Linguagem que aceita números binários pares

Simulando o Reconhecimento: ex1

palavra de entrada

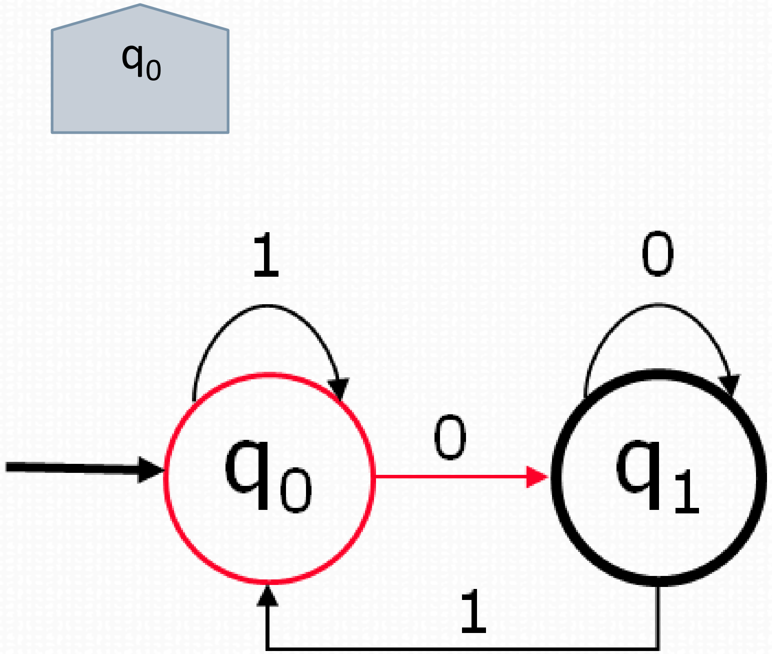


Estado Inicial

Simulando o Reconhecimento: ex1

palavra de entrada

0	1	1	0	1



transição

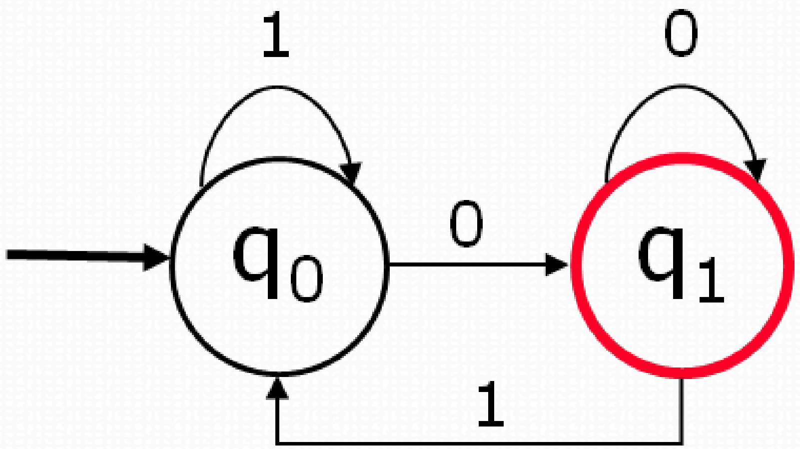
$\underline{\delta}\ (q_0\ ,\ 0) = q_1$

Simulando o Reconhecimento: ex1

palavra de entrada

0	1	1	0	1

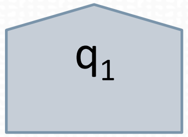
q_1



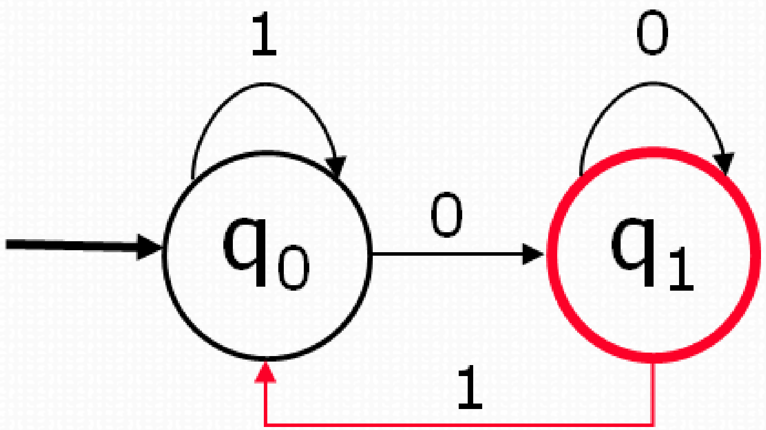
Simulando o Reconhecimento: ex1

palavra de entrada

0	1	1	0	1



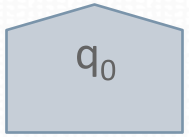
transição
$\underline{\delta}\,(q_1 , 1) = q_0$

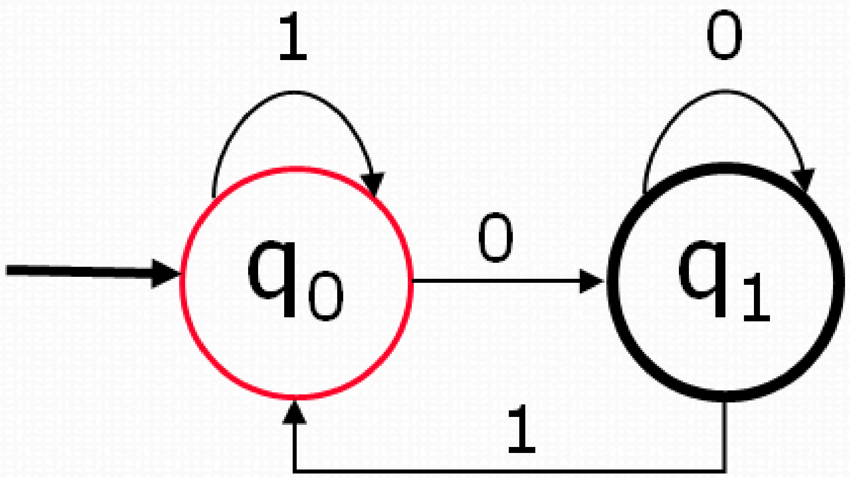


Simulando o Reconhecimento: ex1

palavra de entrada

0	1	1	0	1





Simulando o Reconhecimento: ex1

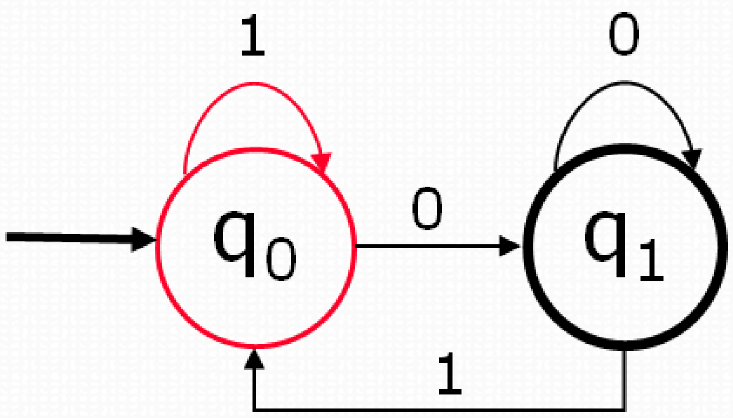
palavra de entrada

0	1	1	0	1

q_0

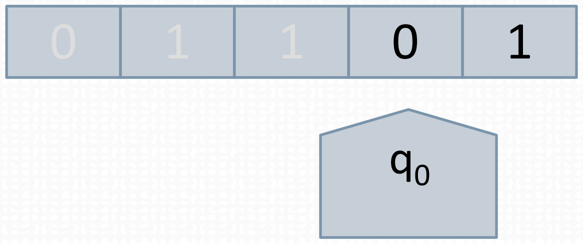
transição

$\underline{\delta}\,(q_0\,,\,1) = q_0$



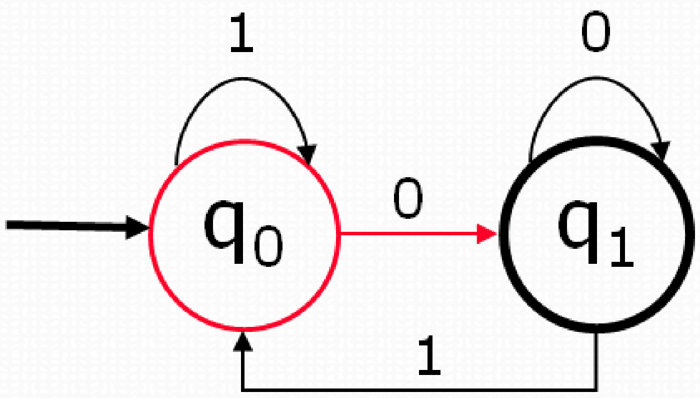
Simulando o Reconhecimento: ex1

palavra de entrada



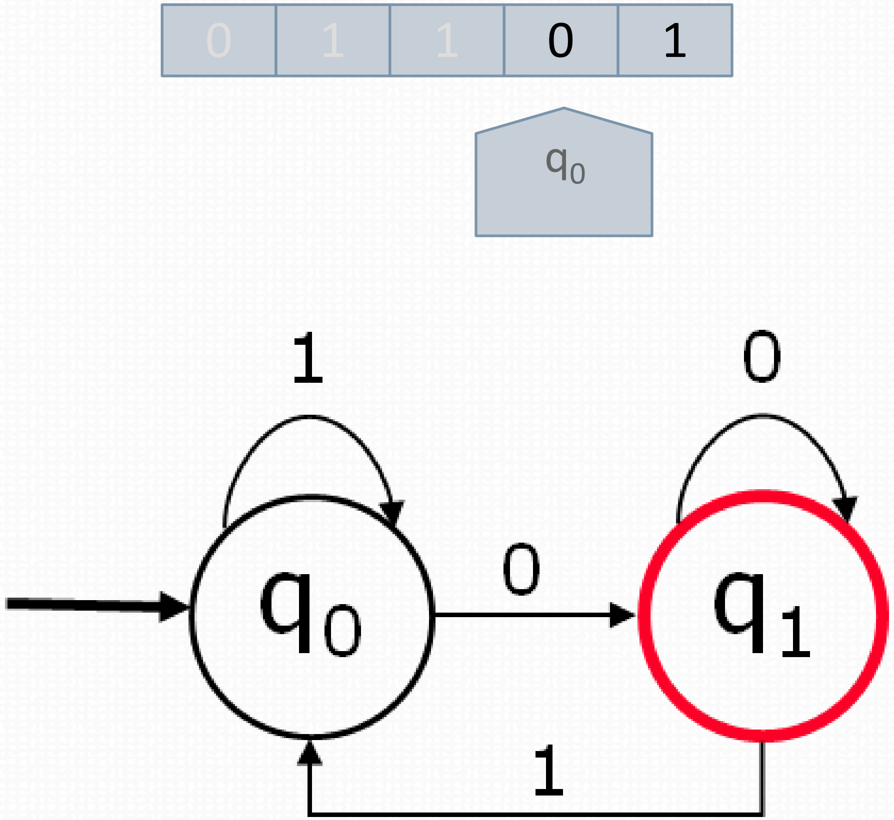
transição

$\underline{\delta}$ $(q_0 , 0) = q_1$

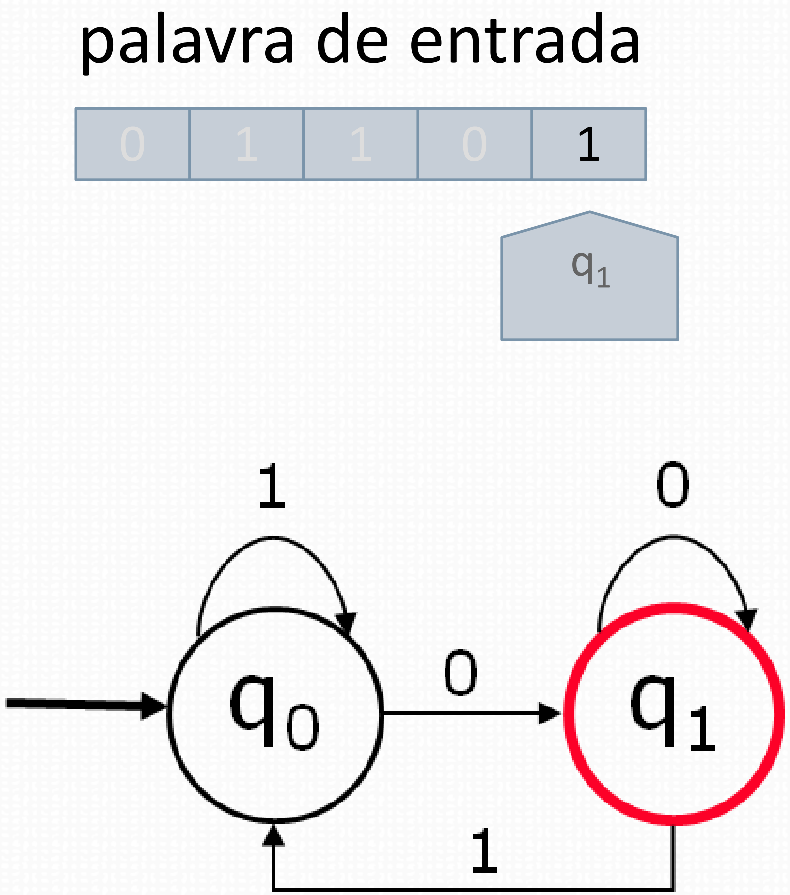


Simulando o Reconhecimento: ex1

palavra de entrada



Simulando o Reconhecimento: ex1

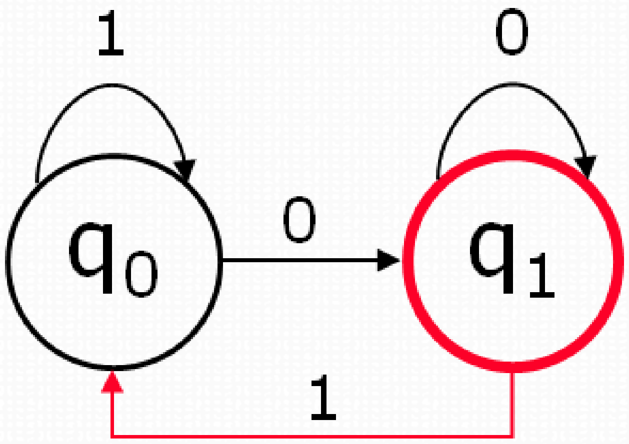


Simulando o Reconhecimento: ex1

palavra de entrada

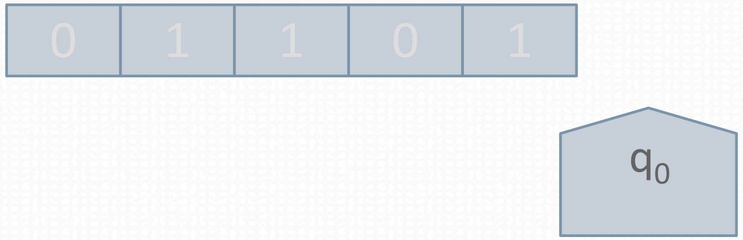
transição

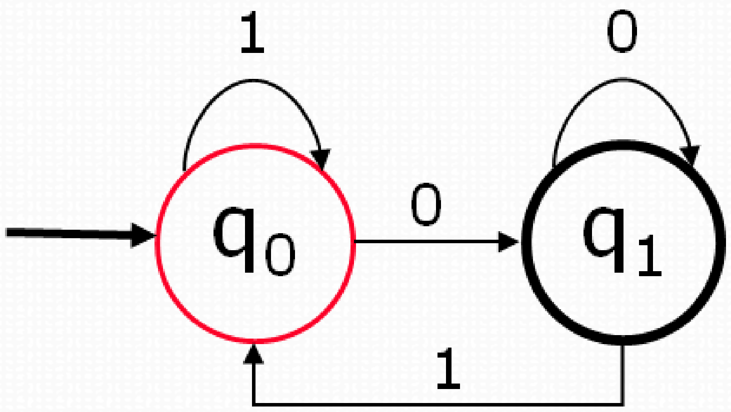
$\underline{\delta}\ (q_1\ ,\ 1) = q_0$



Simulando o Reconhecimento: ex1

palavra de entrada





transição

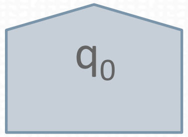
$\underline{\delta}\,(q_1\,,\,1) = q_0$

A palavra “01101” não pertence a linguagem

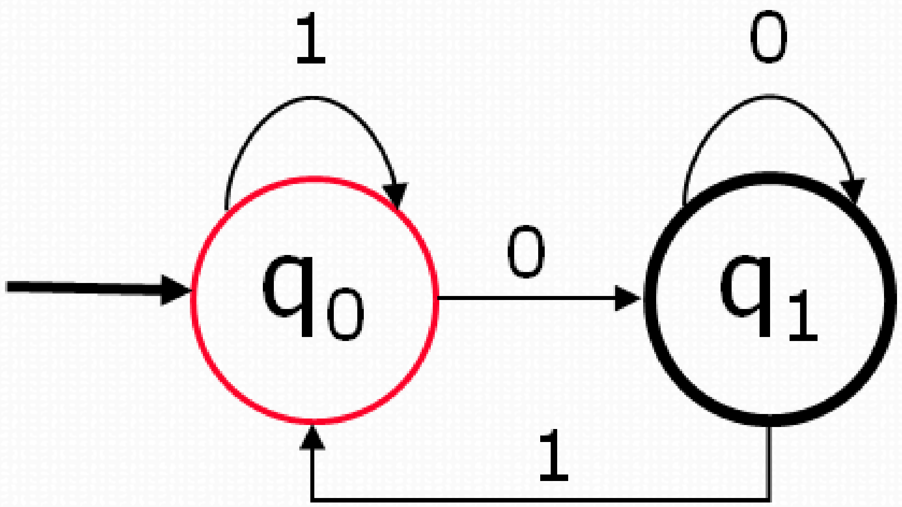
Simulando o Reconhecimento: ex2

palavra de entrada

0	1	1	0



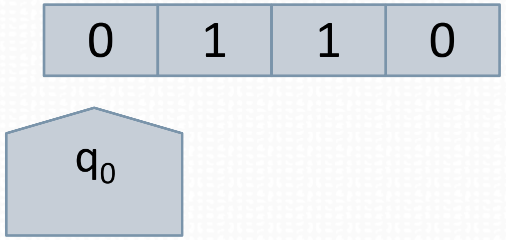
Estado Inicial



UNIVALI

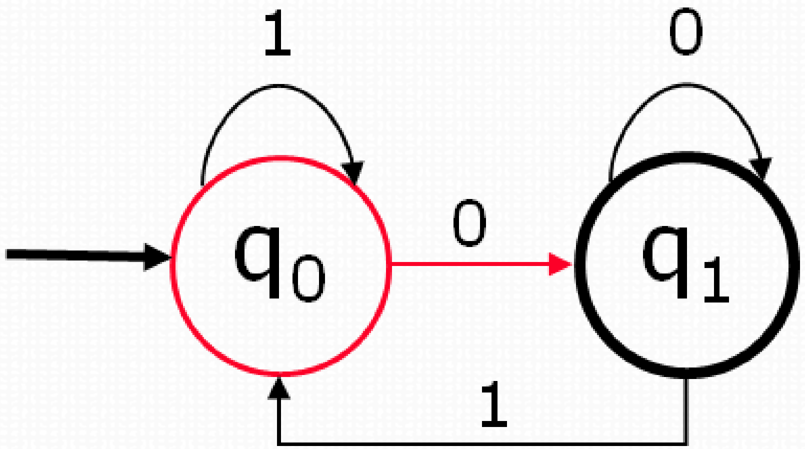
Simulando o Reconhecimento: ex2

palavra de entrada



transição

$\underline{\delta}\ (q_0 , 0) = q_1$

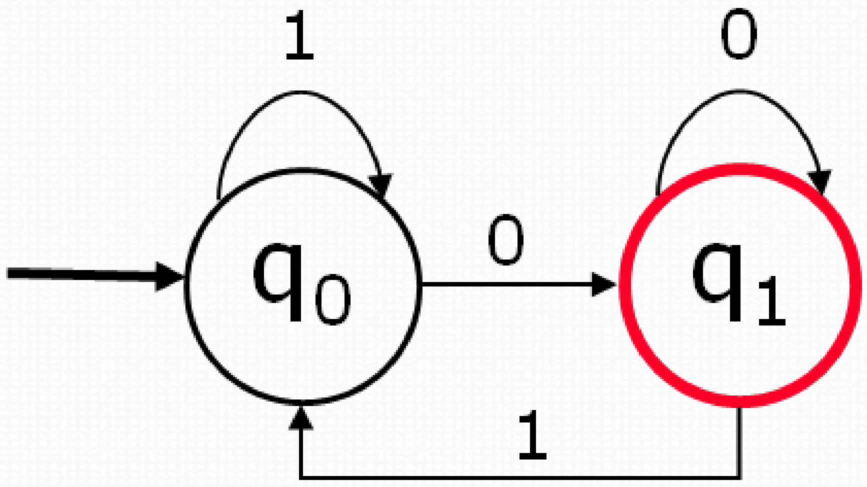


Simulando o Reconhecimento: ex2

palavra de entrada

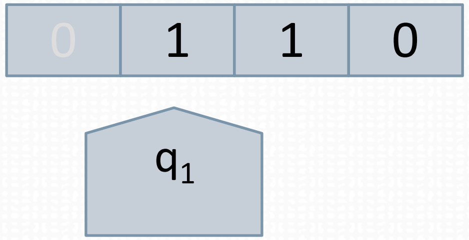
0	1	1	0

q_1



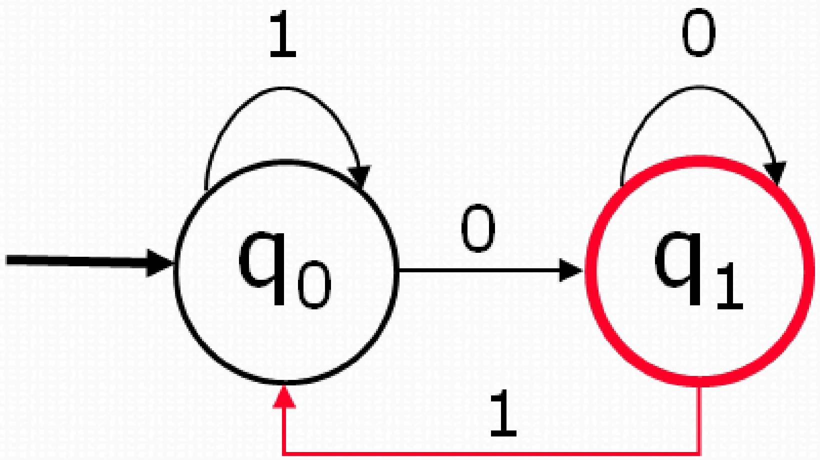
Simulando o Reconhecimento: ex2

palavra de entrada



transição

$\underline{\delta}\,(q_1, 1) = q_0$

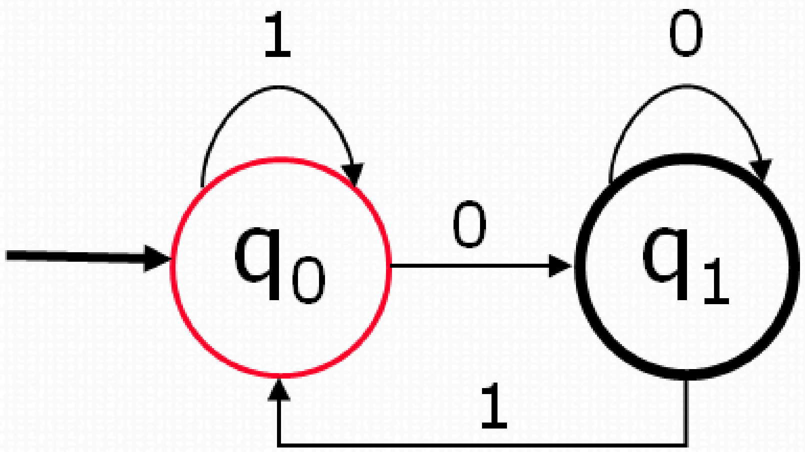


Simulando o Reconhecimento: ex2

palavra de entrada

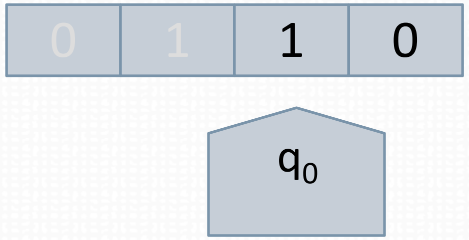
0	1	1	0

q_0



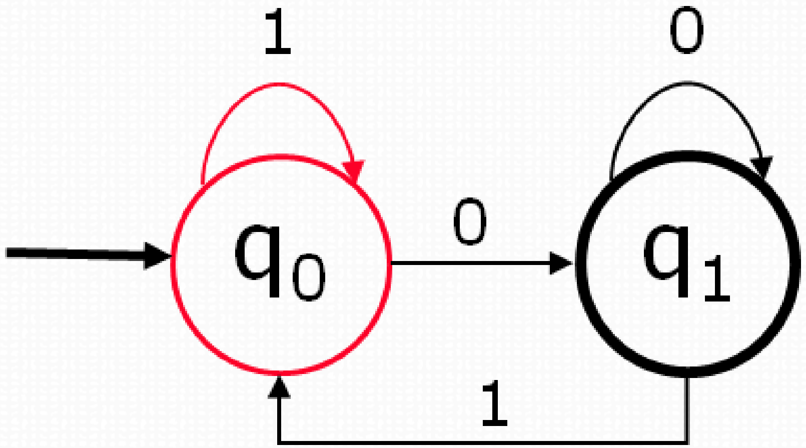
Simulando o Reconhecimento: ex2

palavra de entrada



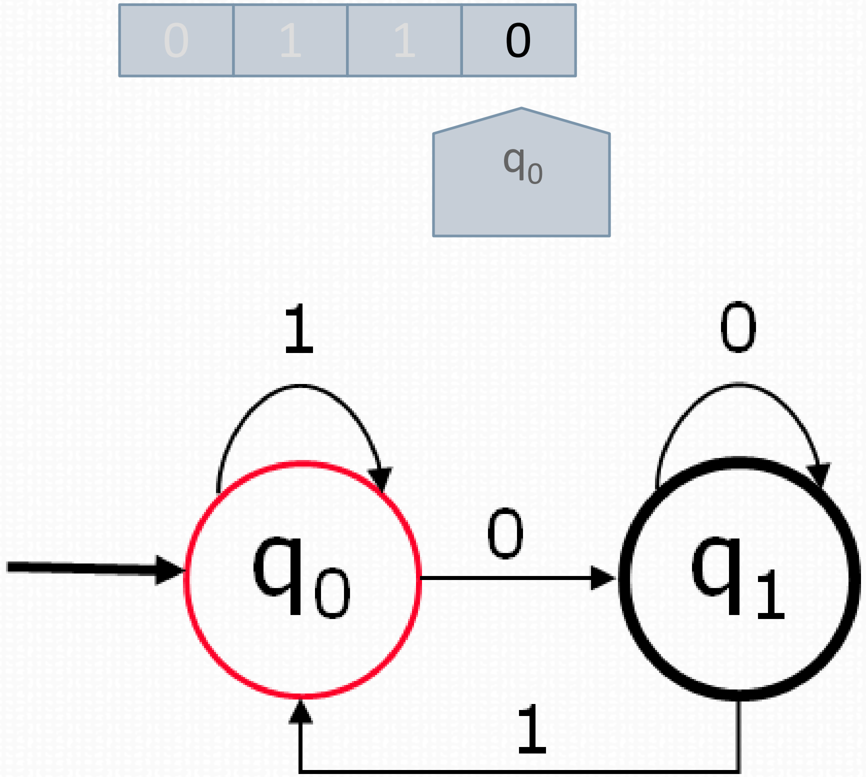
transição

$\underline{\delta}\ (q_0, 1) = q_0$



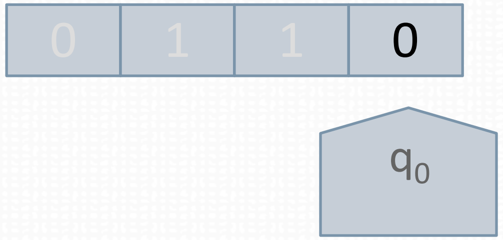
Simulando o Reconhecimento: ex2

palavra de entrada



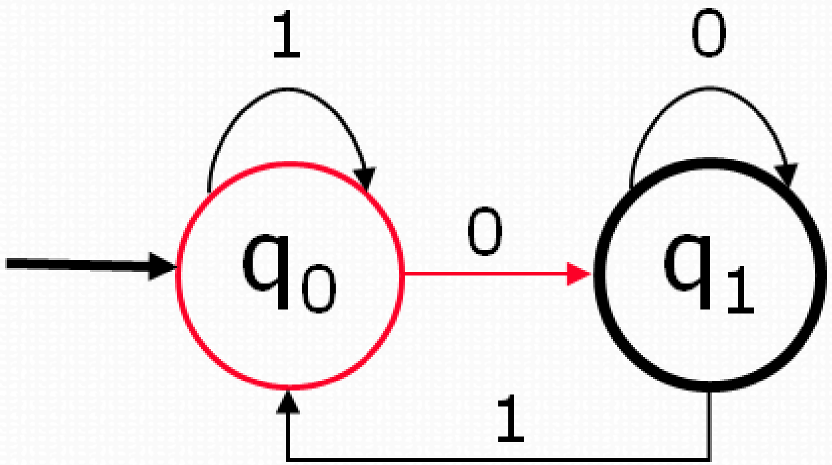
Simulando o Reconhecimento: ex2

palavra de entrada



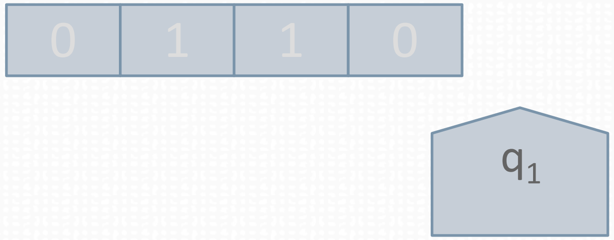
transição

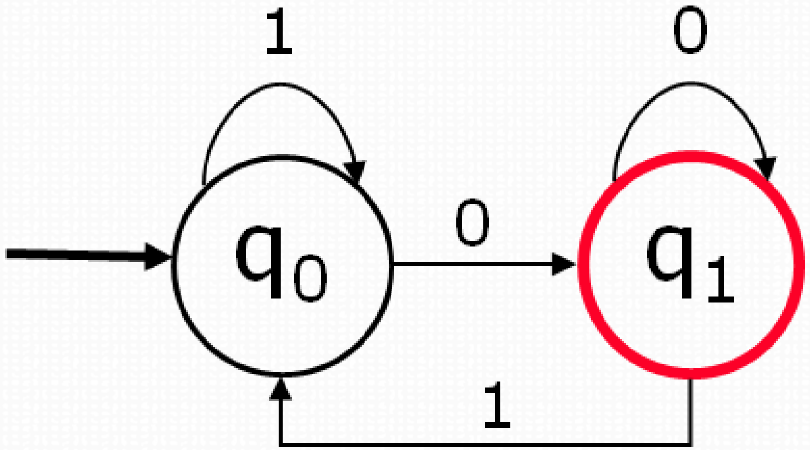
$\delta\,(q_0\,,\,0) = q_1$



Simulando o Reconhecimento: ex2

palavra de entrada





A palavra “0110” pertence a linguagem

Parada do Autômato

Um AFD para nas seguintes situações:

Após ler o último símbolo da fita

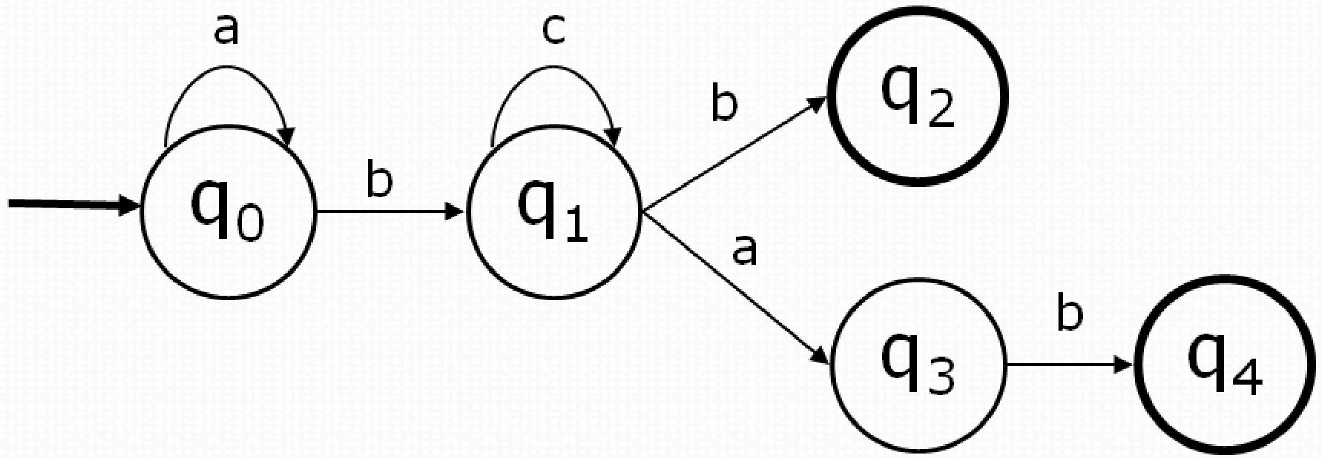
- se o estado pertence a F, aceita a palavra
- se o estado não pertence a F, rejeita a palavra

Quando não existe transição definida para o símbolo lido

- neste caso a palavra é rejeitada

Exemplo de transição indefinida

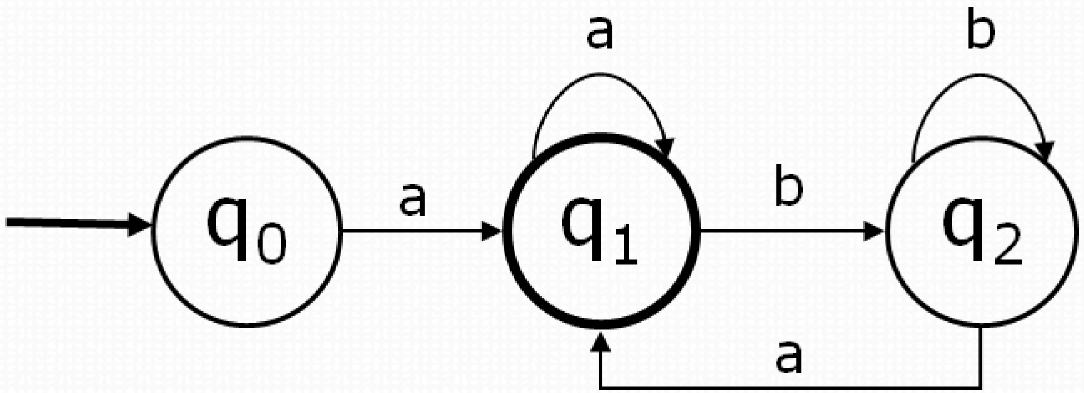
$M = (\{a,b,c\}, \{q_0, q_1, q_2, q_3, q_4\}, \delta, q_0, \{q_2, q_4\})$



Palavras rejeitadas: {“ac”; “bac”; “bbc”;...}

Exemplos

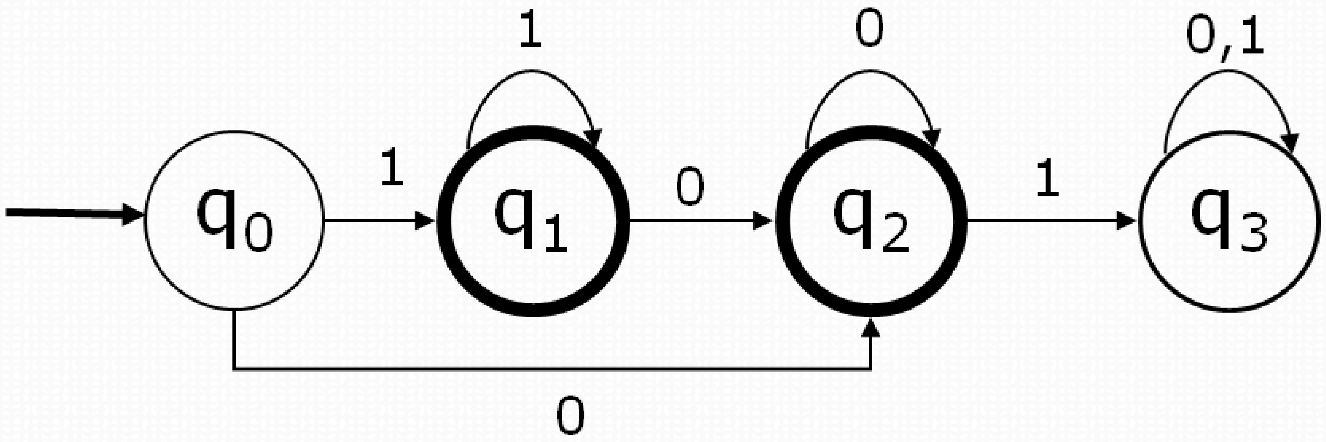
$$M = (\{a,b\}, \{q_0, q_1, q_2\}, \delta, q_0, q_1)$$



Linguagem das sentenças que iniciam e terminam por "a" (possuem "a" como prefixo e sufixo)

Exemplos

$$M = (\{0,1\}, \{q_0, q_1, q_2, q_3\}, \delta, q_0, \{q_1, q_2\})$$

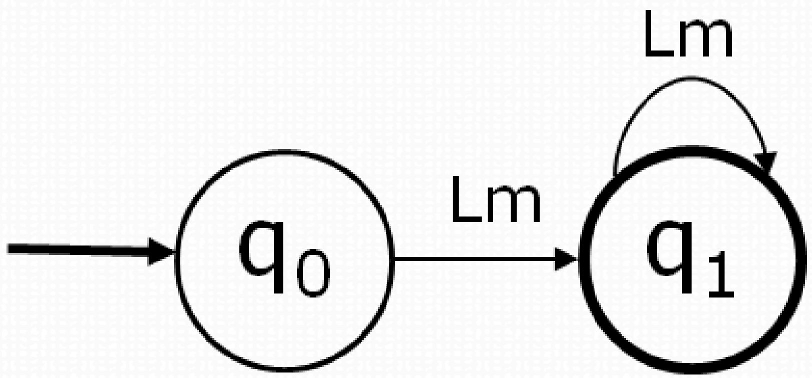


Palavras que não contém a subpalavra "01"

Estado q_2 = estado onde um zero foi recebido

Exemplos

Lm = {a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,x,y,w,z}



Palavras com 1 ou mais letras

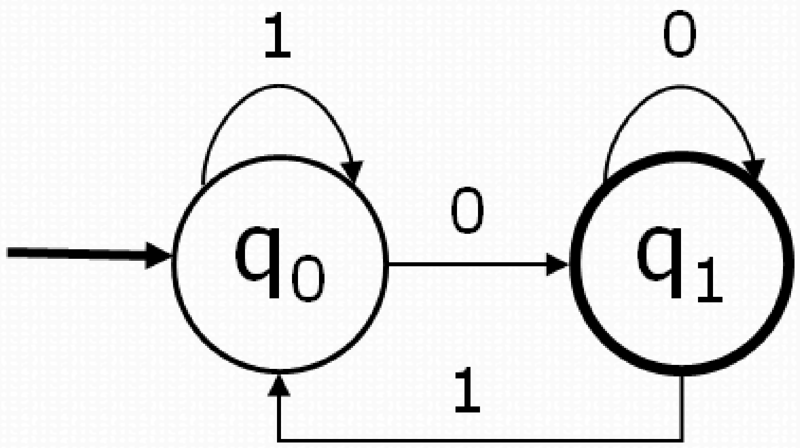
Exercícios

Apresente a definição formal e o grafo da função de transição para as seguintes linguagens:

- números binários ímpares ($\Sigma = \{0,1\}$)
- palavras que possuem “ab” como prefixo ($\Sigma = \{a,b\}$)
- números naturais ($\Sigma = \{0,1,2,3,4,5,6,7,8,9\}$)
- números reais ($\Sigma = \{0,1,2,3,4,5,6,7,8,9,.\}$)
- palavras que não contém “001” como subpalavra

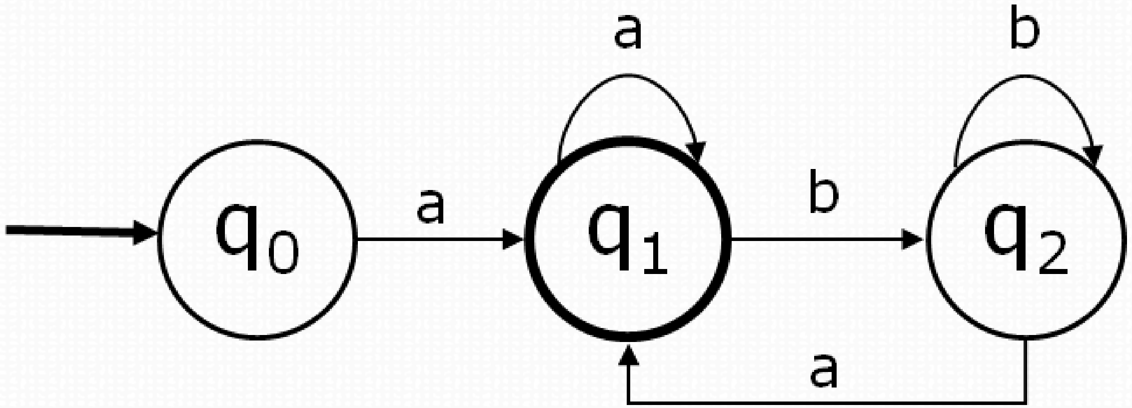
Tabelas de Transição

	δ	0	1
$\rightarrow$	q_0	q_1	q_0
$*$	q_1	q_1	q_0



Tabelas de Transição

	δ	a	b
$\rightarrow$	q_0	q_1	-
$*$	q_1	q_1	q_2
	q_2	q_1	q_2



Gramática ⇦⇨ AF

Gramáticas Regulares

- Conversão GR em AFND-ε

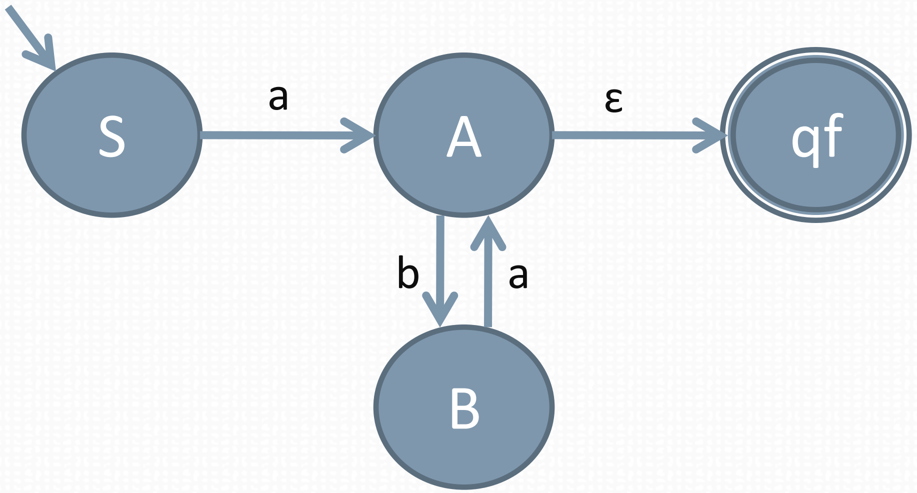
  - Considerando a seguinte GLUD
    - $S \rightarrow aA$
    - $A \rightarrow bB \mid \varepsilon$
    - $B \rightarrow aA$

Produção	Transição
$S \rightarrow aA$	$\delta(S, a) = A$
$A \rightarrow bB$	$\delta(A, b) = B$
$A \rightarrow \varepsilon$	$\delta(A, \varepsilon) = qf$
$B \rightarrow aA$	$\delta(B, a) = A$

Gramáticas Regulares

- Conversão GR em AFND-ε

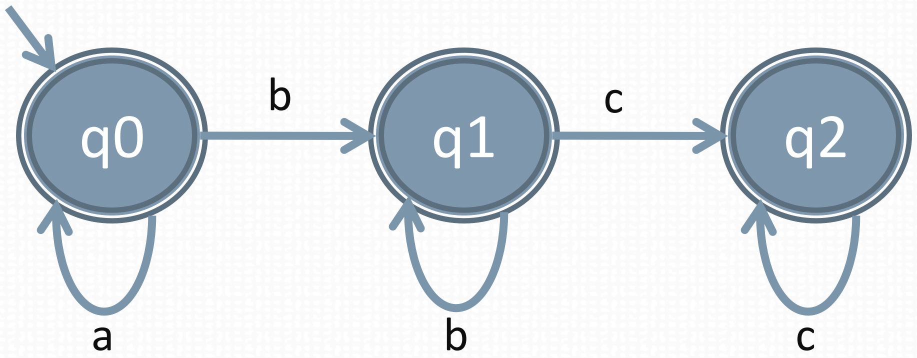
Produção	Transição
S→ aA	δ (S, a) = A
A→ bB	δ (A, b) = B
A→ ε	δ (A, ε) = qf
B →aA	δ (B, a) = A



UNIVALI

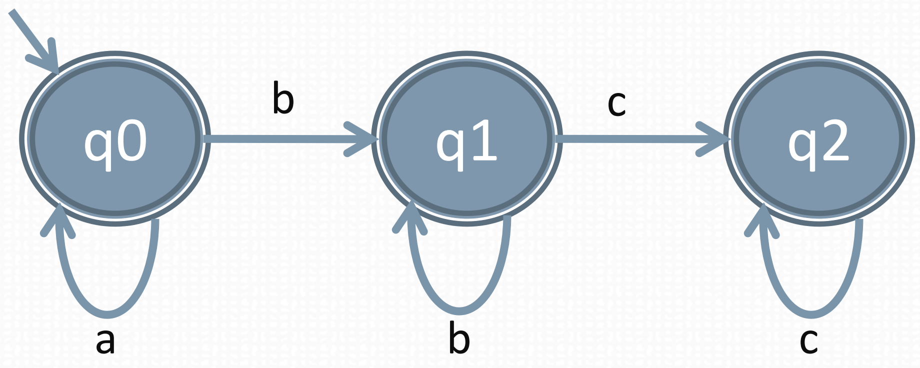
Gramáticas Regulares

- Conversão AFND em GR



Gramáticas Regulares

- Conversão AFND em GR



Transição	Produção
-	S→ q0
-	q0→ ε
-	q1→ ε
-	q2→ ε
δ (q0, a) = q0	q0→aq0
δ (q0, b) = q1	q0→bq1
δ (q1, b) = q1	q1→bq1
δ (q1, c) = q2	q1→cq2
δ (q2, c) = q2	q2→cq2