ACH2043
INTRODUÇÃO À TEORIA DA COMPUTAÇÃO

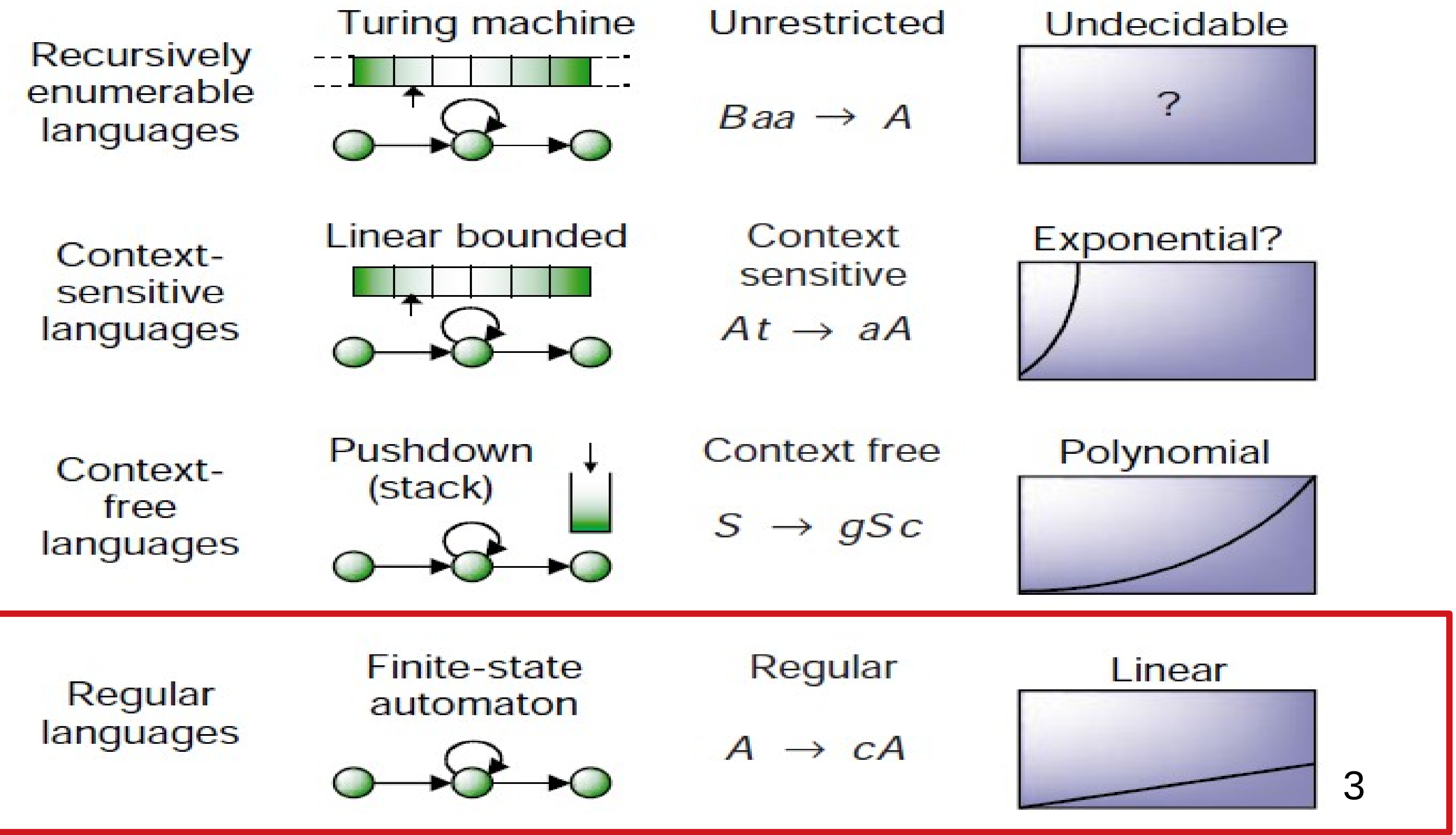
Aula 3

Autômatos Finitos Determinísticos

Profa. Ariane Machado Lima
ariane.machado@usp.br

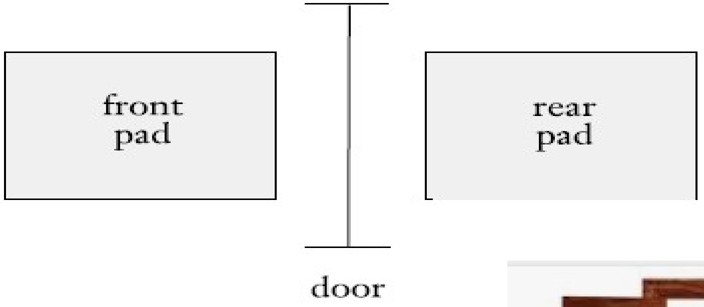
Aula passada

Linguagens, modelos computacionais (dispositivos, gramáticas) e suas complexidades



Autômatos finitos

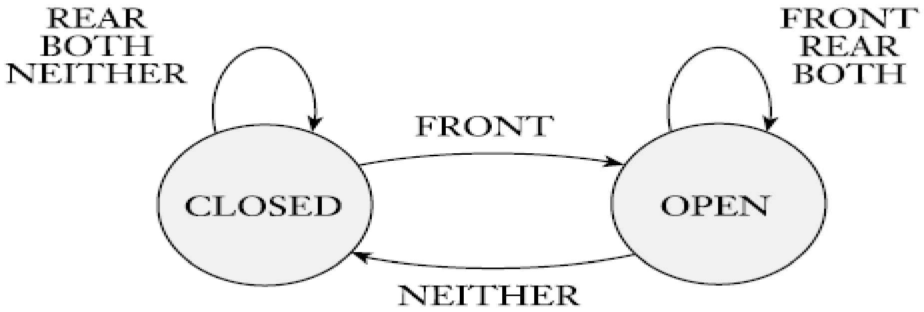
- O exemplo de um controlador de portas que abre (para trás) só para quem está chegando
- Um sensor na frente e outro atrás
- Quais são os estados possíveis da porta?
  - Aberta / Fechada



- Entradas possíveis vindas do sensor (presença de pessoas):
  - FRONT (tem gente só na frente)
  - REAR (tem gente só atrás)
  - BOTH (tem gente na frente e atrás)
  - NEITHER (ninguém na frente nem atrás)

Autômatos finitos

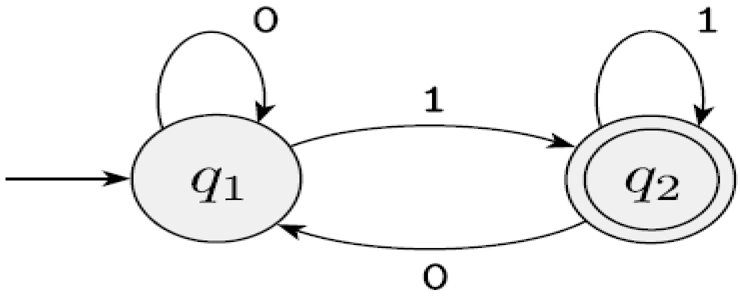
- O exemplo de um controlador de portas



	FRONT	REAR	BOTH	NEITHER
OPEN	OPEN	OPEN	OPEN	CLOSED
CLOSED	OPEN	CLOSED	CLOSED	CLOSED

Autômatos finitos

- Um AF pode ser definido por um diagrama de estados



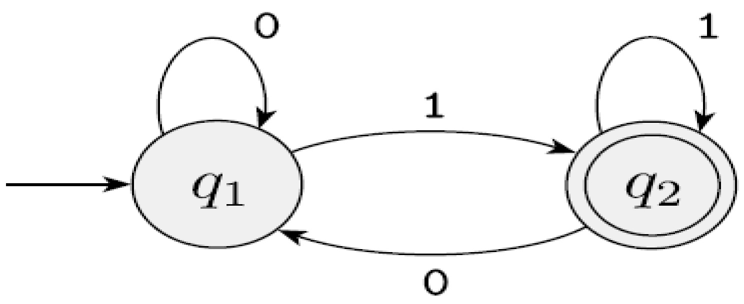
Círculos: estados

Círculos duplos: estados finais ou de aceitação

Seta sem início (somente uma): indica quem é o estado inicial

Setas entre estados: define a transição entre dois estados após a leitura do símbolo que está sobre a seta

Autômatos finitos



- Precisa guardar em memória:
  - Estado atual

RESPOSTA CORRENTE!!!

- Ex: reconhecer strings binárias (compostas por 0's e 1's) que terminem com 1, com tamanho pelo menos 1
  - 1, 01, 11, 001, 00000001, 1101111, ...

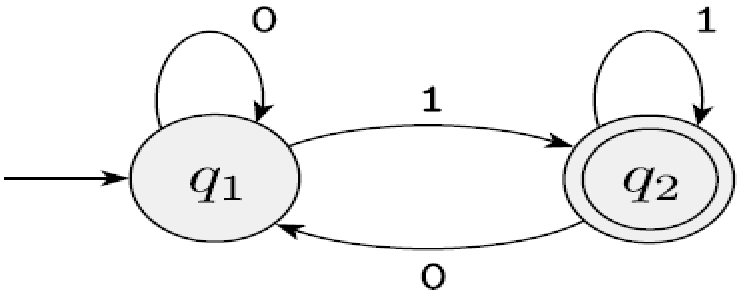
Processo de reconhecimento:

Dados um autômato M e uma cadeia w:

- comece pelo estado inicial
- faça a transição de estados a cada símbolo lido (da cadeia de entrada w)
- ao finalizar a leitura, se M parou em um estado de aceitação aceite, e rejeite caso contrário

Diagrama de estados:

M1



Exemplos de cadeias w aceitas pelo autômato M1:
1, 01, 11, 001, 00000001, 1101111, …

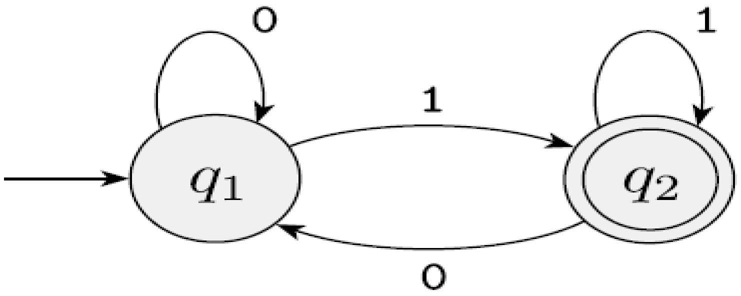
Exemplos de cadeias w NÃO aceitas pelo autômato M1:
0, 10, 110, 000, 0001010, 1111110, ε, ...

Cadeia vazia ("")

Autômatos finitos

- Diagrama de estados

M1



O autômato:
- Aceita uma cadeia
- Reconhece uma linguagem

Resumo

- Um **alfabeto** Σ é um conjunto de símbolos
- Uma **cadeia** w é uma sequência (concatenação) de símbolos de um alfabeto Σ, incluindo a **cadeia vazia ε** (w є Σ*)
- Uma **linguagem** L é **um conjunto de cadeias** sobre um alfabeto Σ (L é subconjunto de Σ*)
- A **linguagem** L reconhecida por um autômato M (denotado L(M)) é o conjunto das **cadeias** (de símbolos de entrada) aceitas pelo autômato

Exercício

Projete um AFD que reconheça cadeias binárias (compostas por 0’s e 1’s) que comecem e terminem com zero, com tamanho pelo menos 1

0, 00, 010, 000000, 0101110, ...

Profa. Ariane Machado Lima

Aula de Hoje

Autômatos finitos

- Podem ser (os principais tipos):
  - Determinísticos
  - Não determinísticos

Um é mais eficiente, o outro é mais fácil de projetar….

Autômatos finitos

- Podem ser (os principais tipos):
  - Determinísticos
  - Não determinísticos

Autômatos Finitos Determinísticos

- Definição formal:

Um ***autômato finito*** é uma 5-upla $(Q, \Sigma, \delta, q_0, F)$, onde

1. Q é um conjunto finito conhecido como os ***estados***,
2. Σ é um conjunto finito chamado o ***alfabeto***,
3. $\delta: Q \times \Sigma \longrightarrow Q$ é a ***função de transição***,[1]
4. $q_0 \in Q$ é o ***estado inicial***, e
5. $F \subseteq Q$ é o ***conjunto de estados de aceitação***.[2]

Autômatos Finitos Determinísticos

- Definição formal:

Um ***autômato finito*** é uma 5-upla $(Q, \Sigma, \delta, q_0, F)$, onde

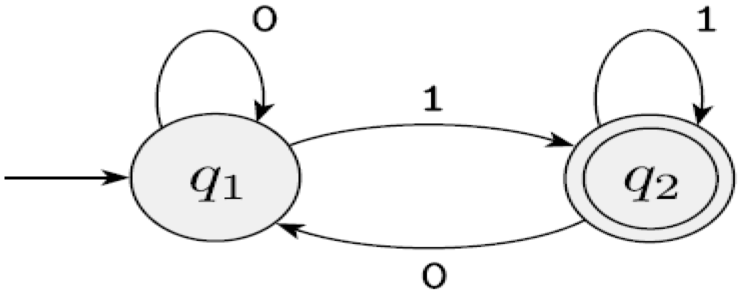
1. Q é um conjunto finito conhecido como os ***estados***,
2. Σ é um conjunto finito chamado o ***alfabeto***,
3. $\delta\colon Q \times \Sigma \longrightarrow Q$ é a ***função de transição***,[1]
4. $q_0 \in Q$ é o ***estado inicial***, e
5. $F \subseteq Q$ é o ***conjunto de estados de aceitação***.[2]

total (definida para cada ponto do domínio)

potencialmente vazio

Autômatos Finitos Determinísticos (AFD)

- Dado um estado atual e um símbolo de entrada sabemos exatamente para onde ir (está determinado)



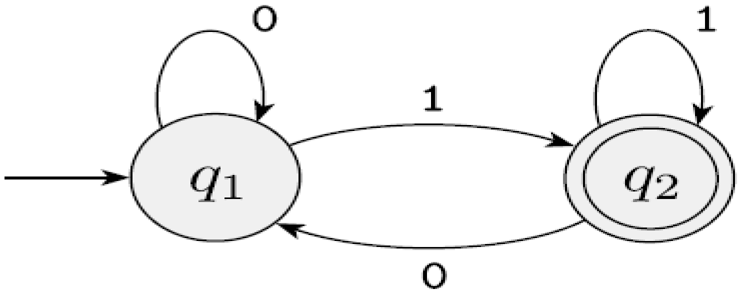
Um *autômato finito* é uma 5-upla $(Q, \Sigma, \delta, q_0, F)$, onde

1. Q é um conjunto finito conhecido como os *estados*,
2. Σ é um conjunto finito chamado o *alfabeto*,
3. $\delta: Q \times \Sigma \longrightarrow Q$ é a *função de transição*,[1]
4. $q_0 \in Q$ é o *estado inicial*, e
5. $F \subseteq Q$ é o *conjunto de estados de aceitação*.[2]

Para cada par (estado atual, próximo símbolo) está DETERMINADO qual é o próximo estado

Autômatos Finitos Determinísticos (AFD)

- Dado um estado atual e um símbolo de entrada sabemos exatamente para onde ir (está determinado)



Um *autômato finito* é uma 5-upla $(Q, \Sigma, \delta, q_0, F)$, onde

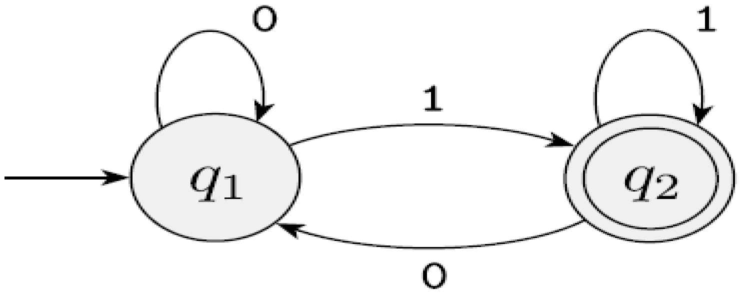
1. Q é um conjunto finito conhecido como os *estados*,
2. Σ é um conjunto finito chamado o *alfabeto*,
3. $\delta : Q \times \Sigma \longrightarrow Q$ é a *função de transição*,[1]
4. $q_0 \in Q$ é o *estado inicial*, e
5. $F \subseteq Q$ é o *conjunto de estados de aceitação*.[2]

Note que para um AFD deve haver, saindo de cada estado, uma aresta para CADA símbolo do alfabeto

Autômatos Finitos Determinísticos (AFD)

- Dado um estado atual e um símbolo de entrada sabemos exatamente para onde ir (está determinado)

Por isso a tabela que define o AFD deve estar totalmente preenchida !!!



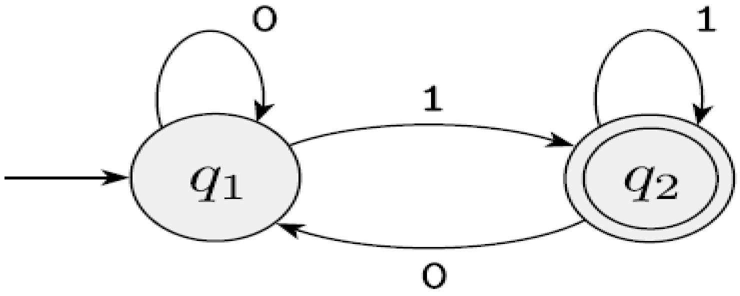
		0	1
→	q1		
←	q2		

Na notação tabular, como se pode perceber, o estado inicial é indicado através do símbolo "→", ao passo que os estados finais são indicados por "←". O símbolo "↔" indica um estado que seja simultaneamente inicial e final.

Autômatos Finitos Determinísticos (AFD)

- Dado um estado atual e um símbolo de entrada sabemos exatamente para onde ir (está determinado)

Por isso a tabela que define o AFD deve estar totalmente preenchida !!!



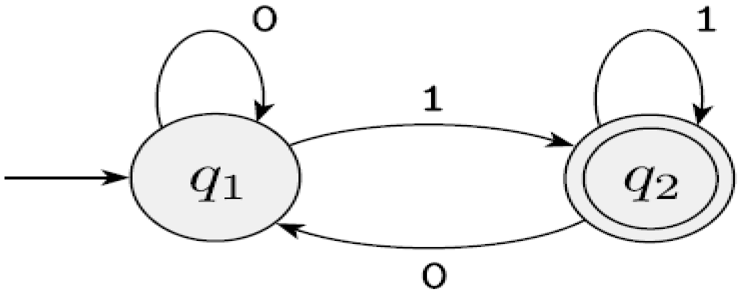
		0	1
→	**q1**	q1	.
←	**q2**	.	.

Na notação tabular, como se pode perceber, o estado inicial é indicado através do símbolo "→", ao passo que os estados finais são indicados por "←". O símbolo "↔" indica um estado que seja simultaneamente inicial e final.

Autômatos Finitos Determinísticos (AFD)

- Dado um estado atual e um símbolo de entrada sabemos exatamente para onde ir (está determinado)

Por isso a tabela que define o AFD deve estar totalmente preenchida !!!



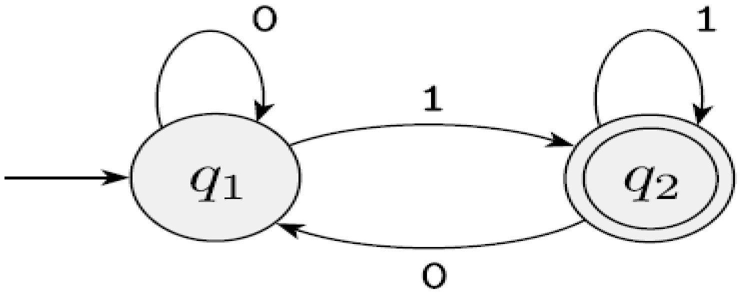
		0	1
→	**q1**	q1	q2
←	**q2**	.	.

Na notação tabular, como se pode perceber, o estado inicial é indicado através do símbolo "→", ao passo que os estados finais são indicados por "←". O símbolo "↔" indica um estado que seja simultaneamente inicial e final.

Autômatos Finitos Determinísticos (AFD)

- Dado um estado atual e um símbolo de entrada sabemos exatamente para onde ir (está determinado)

Por isso a tabela que define o AFD deve estar totalmente preenchida !!!



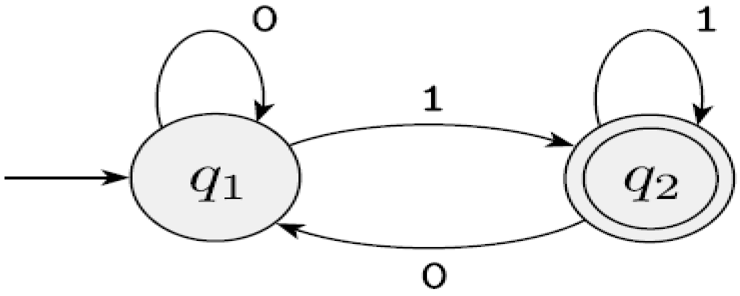
		0	1
→	**q1**	q1	q2
←	**q2**	q1	

Na notação tabular, como se pode perceber, o estado inicial é indicado através do símbolo "→", ao passo que os estados finais são indicados por "←". O símbolo "↔" indica um estado que seja simultaneamente inicial e final.

Autômatos Finitos Determinísticos (AFD)

- Dado um estado atual e um símbolo de entrada sabemos exatamente para onde ir (está determinado)

Por isso a tabela que define o AFD deve estar totalmente preenchida !!!



		0	**1**
→	**q1**	q1	q2
←	**q2**	q1	q2

Na notação tabular, como se pode perceber, o estado inicial é indicado através do símbolo "→", ao passo que os estados finais são indicados por "←". O símbolo "↔" indica um estado que seja simultaneamente inicial e final.

Exercício

Verifique se o diagrama de estados que você implementou como exercício da aula 2 (slide 61 : do autômato que aceita cadeias binárias que comecem e terminem com zero, com tamanho pelo menos 1), atende aos requisitos de um AFD. Se não atender, descreva tal diagrama como um AFD.

Exercício

Projete um AFD que reconheça cadeias binárias (compostas por 0’s e 1’s) que comecem e terminem com zero, com tamanho pelo menos 1

0, 00, 010, 000000, 0101110, ...

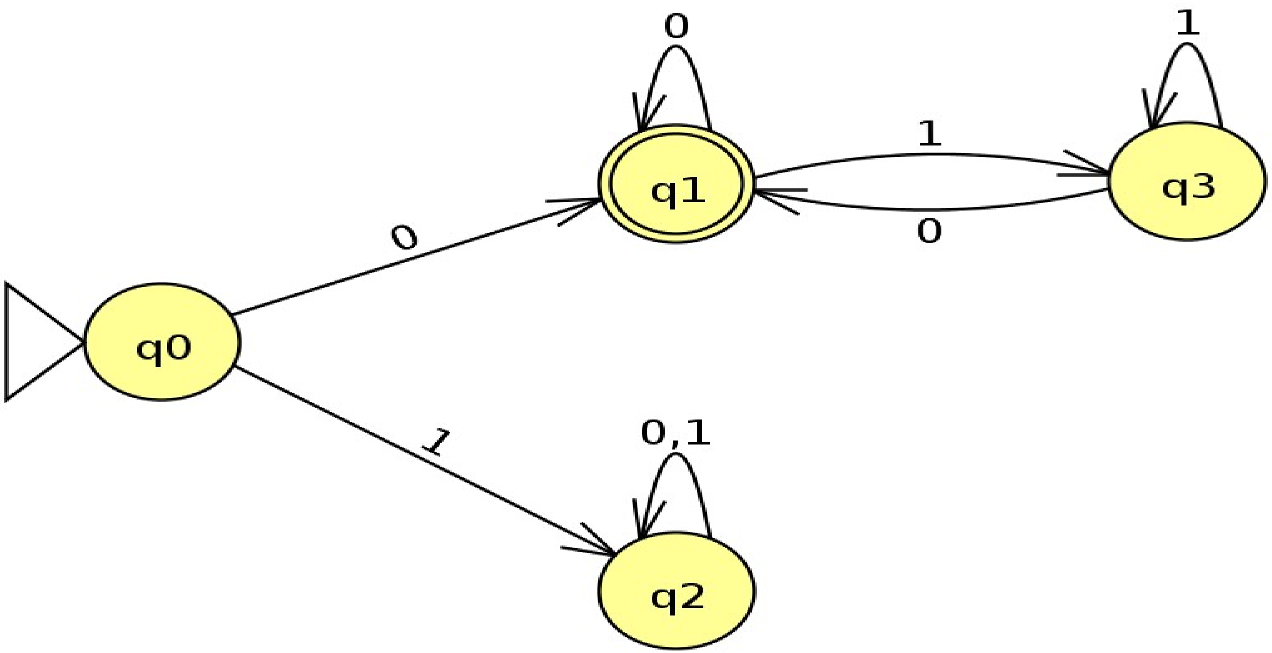
Exercício 1 - resposta

Não mude o slide sem tentar fazer antes!

Exercício 1 - resposta

...que comecem e terminem com zero, com tamanho pelo menos 1

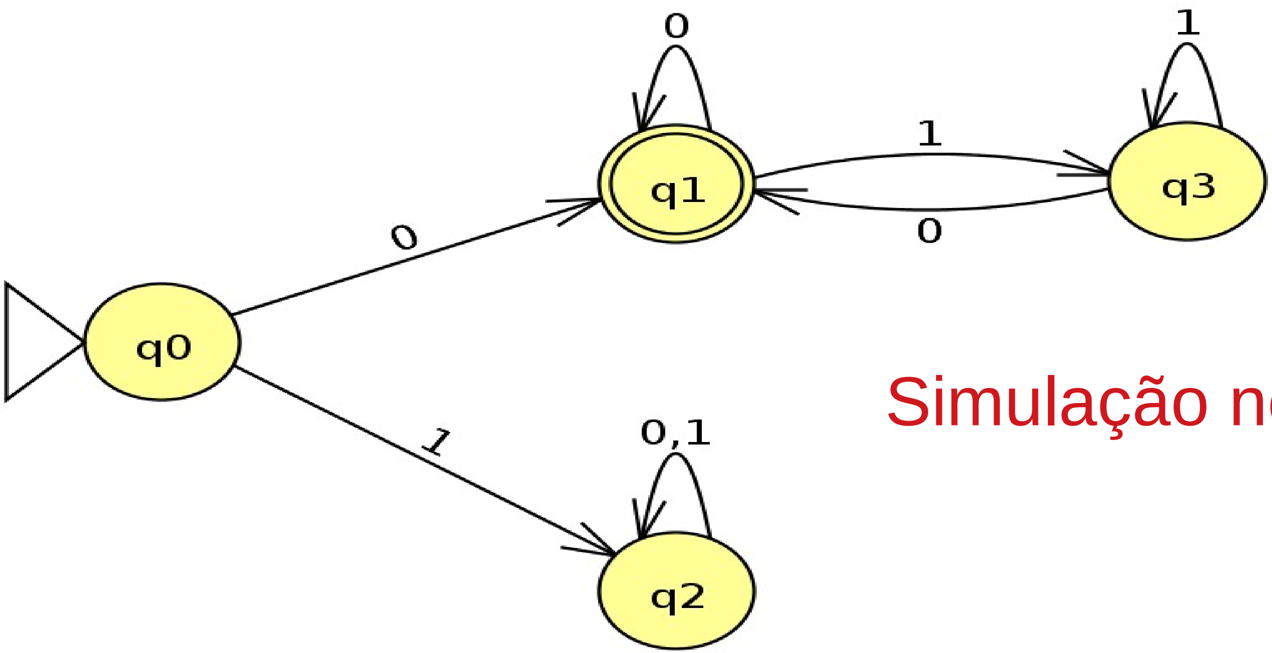
0, 00, 010, 000000, 0101110, ...



Exercício 1 - resposta

...que comecem e terminem com zero, com tamanho pelo menos 1

0, 00, 010, 000000, 0101110, ...



Simulação no JFlap

Qual a complexidade (tempo) de análise de uma cadeia por um AFD?

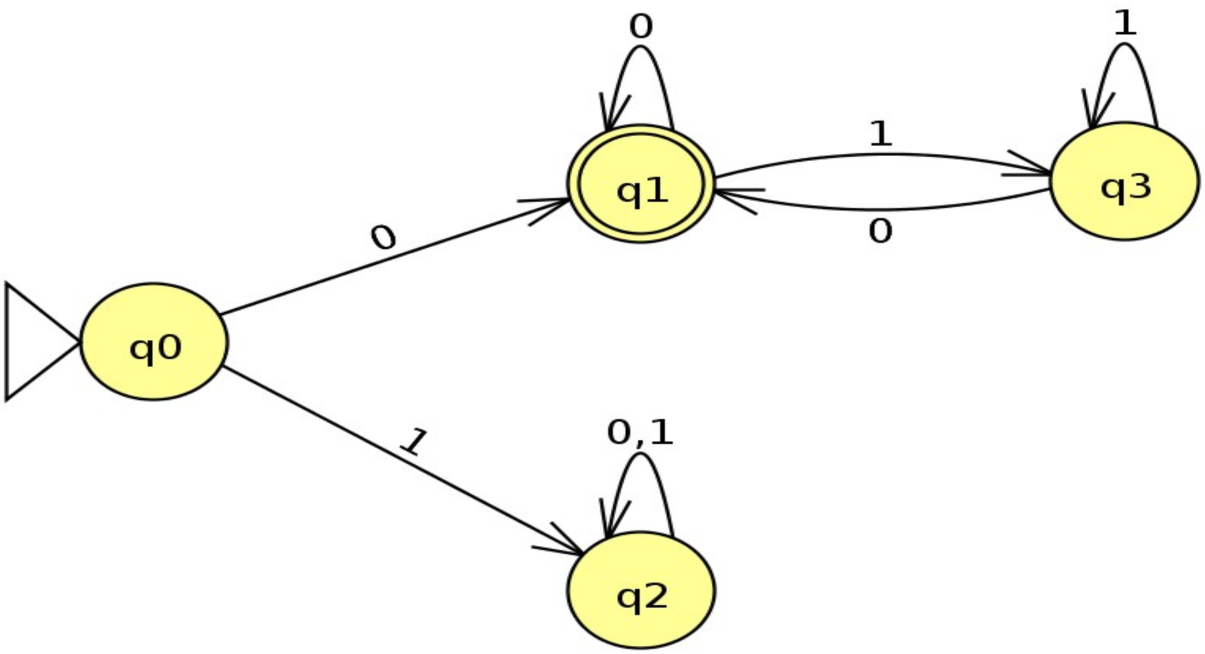
Qual a complexidade (tempo) de análise de uma cadeia por um AFD?

- **O(n)** – n sendo o tamanho da cadeia de entrada. Por quê?

Qual a complexidade (tempo) de análise de uma cadeia por um AFD?

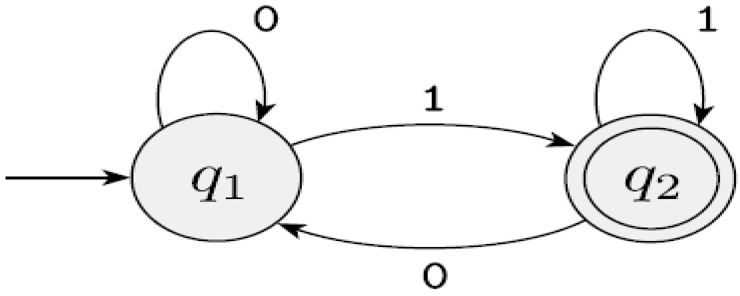
- **O(n)** – n sendo o tamanho da cadeia de entrada

  Porque um símbolo (EXATAMENTE) é lido por transição)



Autômatos Finitos Determinísticos (AFD)

- Qual a definição formal do autômato M1?

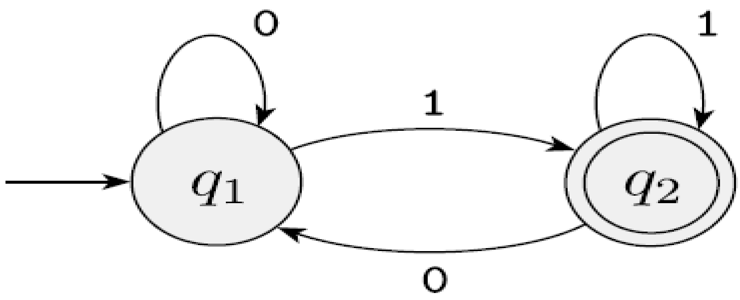


Um ***autômato finito*** é uma 5-upla $(Q, \Sigma, \delta, q_0, F)$, onde

1. Q é um conjunto finito conhecido como os ***estados***,
2. Σ é um conjunto finito chamado o ***alfabeto***,
3. $\delta \colon Q \times \Sigma \longrightarrow Q$ é a ***função de transição***,[1]
4. $q_0 \in Q$ é o ***estado inicial***, e
5. $F \subseteq Q$ é o ***conjunto de estados de aceitação***.[2]

Autômatos Finitos Determinísticos (AFD)

- Qual a definição formal do autômato M1?



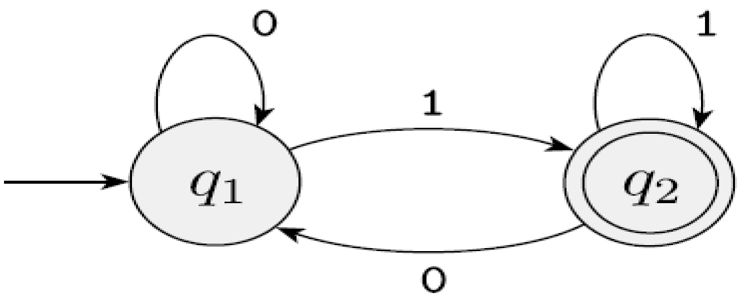
Q =

Um ***autômato finito*** é uma 5-upla $(Q, \Sigma, \delta, q_0, F)$, onde

1. Q é um conjunto finito conhecido como os ***estados***,
2. Σ é um conjunto finito chamado o ***alfabeto***,
3. $\delta \colon Q \times \Sigma \longrightarrow Q$ é a ***função de transição***,[1]
4. $q_0 \in Q$ é o ***estado inicial***, e
5. $F \subseteq Q$ é o ***conjunto de estados de aceitação***.[2]

Autômatos Finitos Determinísticos (AFD)

- Qual a definição formal do autômato M1?



Q = {q1, q2}

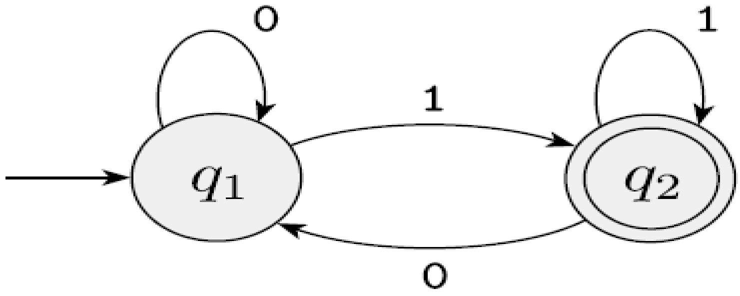
Σ =

Um *autômato finito* é uma 5-upla $(Q, \Sigma, \delta, q_0, F)$, onde

1. Q é um conjunto finito conhecido como os *estados*,
2. Σ é um conjunto finito chamado o *alfabeto*,
3. $\delta \colon Q \times \Sigma \longrightarrow Q$ é a *função de transição*,[1]
4. $q_0 \in Q$ é o *estado inicial*, e
5. $F \subseteq Q$ é o *conjunto de estados de aceitação*.[2]

Autômatos Finitos Determinísticos (AFD)

- Qual a definição formal do autômato M1?



Q = {q1, q2}

Σ = {0 ,1}

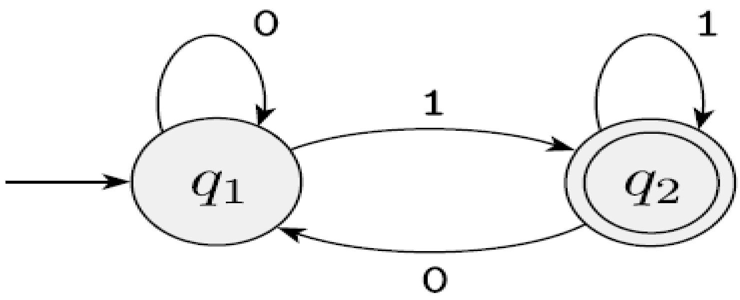
δ

Um ***autômato finito*** é uma 5-upla $(Q, \Sigma, \delta, q_0, F)$, onde

1. Q é um conjunto finito conhecido como os ***estados***,
2. Σ é um conjunto finito chamado o ***alfabeto***,
3. $\delta : Q \times \Sigma \longrightarrow Q$ é a ***função de transição***,[1]
4. $q_0 \in Q$ é o ***estado inicial***, e
5. $F \subseteq Q$ é o ***conjunto de estados de aceitação***.[2]

Autômatos Finitos Determinísticos (AFD)

- Qual a definição formal do autômato M1?



Q = {q1, q2}

Σ = {0 ,1}

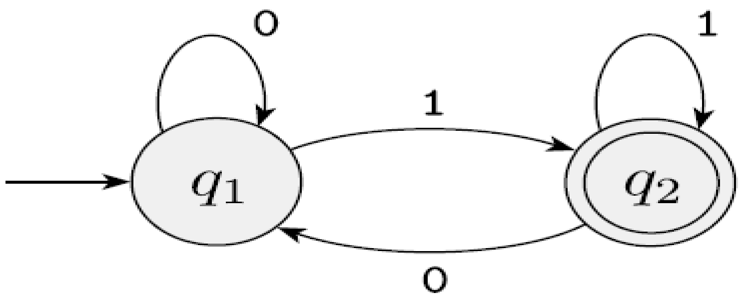
δ(q1, 0) = q1

Um ***autômato finito*** é uma 5-upla $(Q, \Sigma, \delta, q_0, F)$, onde

1. Q é um conjunto finito conhecido como os ***estados***,
2. Σ é um conjunto finito chamado o ***alfabeto***,
3. $\delta \colon Q \times \Sigma \longrightarrow Q$ é a ***função de transição***,[1]
4. $q_0 \in Q$ é o ***estado inicial***, e
5. $F \subseteq Q$ é o ***conjunto de estados de aceitação***.[2]

Autômatos Finitos Determinísticos (AFD)

- Qual a definição formal do autômato M1?



Um ***autômato finito*** é uma 5-upla $(Q, \Sigma, \delta, q_0, F)$, onde

1. Q é um conjunto finito conhecido como os ***estados***,
2. Σ é um conjunto finito chamado o ***alfabeto***,
3. $\delta \colon Q \times \Sigma \longrightarrow Q$ é a ***função de transição***,[1]
4. $q_0 \in Q$ é o ***estado inicial***, e
5. $F \subseteq Q$ é o ***conjunto de estados de aceitação***.[2]

Q = {q1, q2}

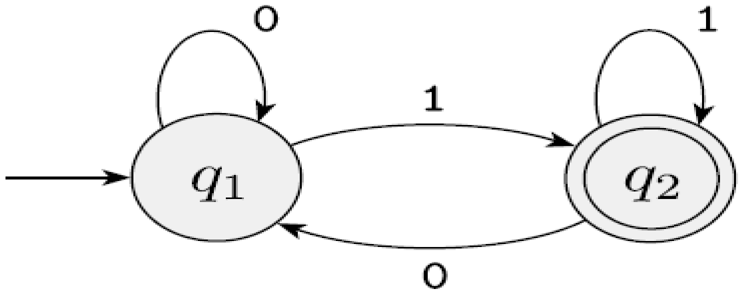
Σ = {0 ,1}

δ(q1, 0) = q1
δ(q1, 1) = q2
δ(q2, 0) = q1
δ(q2, 1) = q2

q0 =

Autômatos Finitos Determinísticos (AFD)

- Qual a definição formal do autômato M1?



Um ***autômato finito*** é uma 5-upla $(Q, \Sigma, \delta, q_0, F)$, onde

1. Q é um conjunto finito conhecido como os ***estados***,
2. Σ é um conjunto finito chamado o ***alfabeto***,
3. $\delta \colon Q \times \Sigma \longrightarrow Q$ é a ***função de transição***,[1]
4. $q_0 \in Q$ é o ***estado inicial***, e
5. $F \subseteq Q$ é o ***conjunto de estados de aceitação***.[2]

Q = {q1, q2}

Σ = {0 ,1}

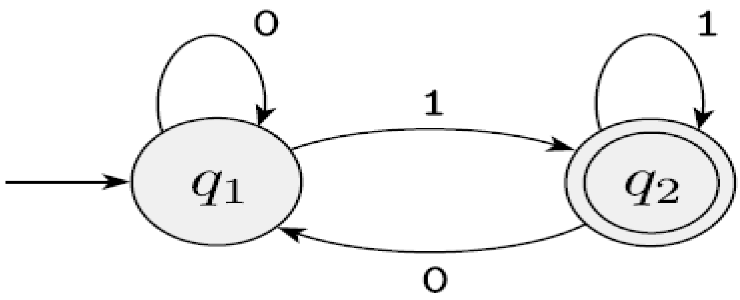
δ(q1, 0) = q1
δ(q1, 1) = q2
δ(q2, 0) = q1
δ(q2, 1) = q2

q0 = q1

F =

Autômatos Finitos Determinísticos (AFD)

- Qual a definição formal do autômato M1?



Um ***autômato finito*** é uma 5-upla $(Q, \Sigma, \delta, q_0, F)$, onde

1. Q é um conjunto finito conhecido como os ***estados***,
2. Σ é um conjunto finito chamado o ***alfabeto***,
3. $\delta \colon Q \times \Sigma \longrightarrow Q$ é a ***função de transição***,[1]
4. $q_0 \in Q$ é o ***estado inicial***, e
5. $F \subseteq Q$ é o ***conjunto de estados de aceitação***.[2]

Q = {q1, q2}

Σ = {0 ,1}

δ(q1, 0) = q1
δ(q1, 1) = q2
δ(q2, 0) = q1
δ(q2, 1) = q2

q0 = q1

F = {q2}

Definição formal de computação

Seja $M = (Q, \Sigma, \delta, q_0, F)$ um autômato finito e suponha que $w = w_1 w_2 \cdots w_n$ seja uma cadeia onde cada w_i é um membro do alfabeto Σ. Então M ***aceita*** w se existe uma seqüência de estados $r_0, r_1, \ldots, r_n$ em Q com três condições:

Definição formal de computação

Seja $M = (Q, \Sigma, \delta, q_0, F)$ um autômato finito e suponha que $w = w_1 w_2 \cdots w_n$ seja uma cadeia onde cada w_i é um membro do alfabeto Σ. Então M ***aceita*** w se existe uma seqüência de estados $r_0, r_1, \ldots, r_n$ em Q com três condições:

1. $r_0 = q_0$,

2.

3. $r_n \in F$.

Profa. Ariane Machado Lima

Definição formal de computação

Seja $M = (Q, \Sigma, \delta, q_0, F)$ um autômato finito e suponha que $w = w_1 w_2 \cdots w_n$ seja uma cadeia onde cada w_i é um membro do alfabeto Σ. Então M ***aceita*** w se existe uma seqüência de estados $r_0, r_1, \ldots, r_n$ em Q com três condições:

1. $r_0 = q_0$,

2. $\delta(r_i, w_{i+1}) = r_{i+1}$, para $i = 0, \ldots, n-1$, e

3. $r_n \in F$.

Definição formal de computação

Seja $M = (Q, \Sigma, \delta, q_0, F)$ um autômato finito e suponha que $w = w_1 w_2 \cdots w_n$ seja uma cadeia onde cada w_i é um membro do alfabeto Σ. Então M ***aceita*** w se existe uma seqüência de estados $r_0, r_1, \ldots, r_n$ em Q com três condições:

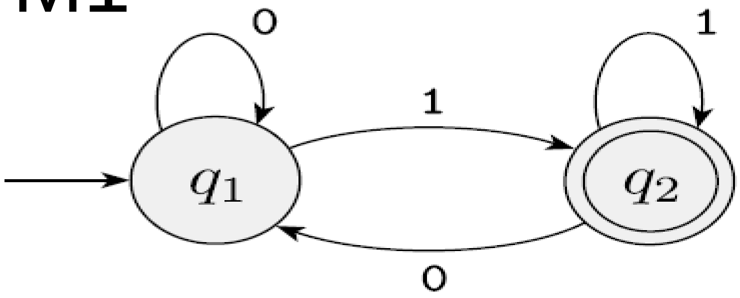
1. $r_0 = q_0$,
2. $\delta(r_i, w_{i+1}) = r_{i+1}$, para $i = 0, \ldots, n-1$, e
3. $r_n \in F$.

Um autômato:

- **aceita** ou não aceita uma **cadeia**
- **reconhece** ou não reconhece uma **linguagem**

Autômatos finitos determinísticos

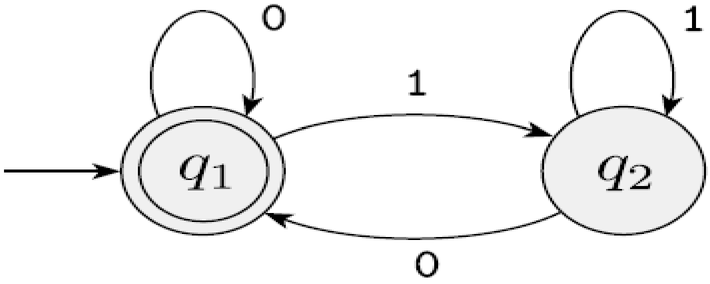
M1



L(M1) = {w ϵ {0,1}* | w termina com 1}

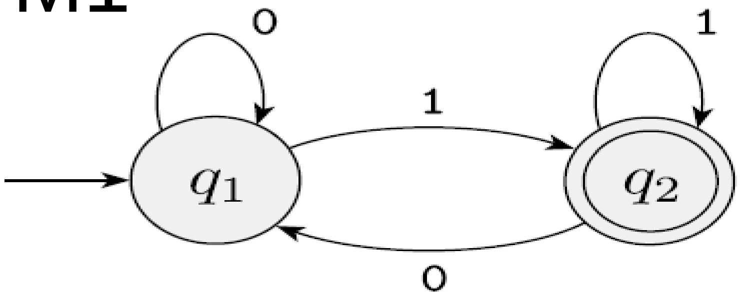
- Que linguagem o autômato M2 reconhece? (apenas mudou o estado final)

M2



Autômatos finitos determinísticos

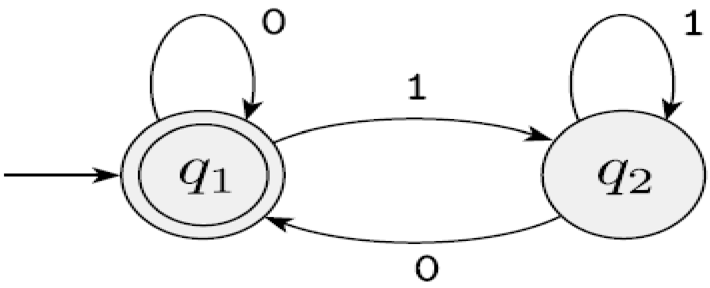
M1



L(M1) = {w ϵ {0,1}* | w termina com 1}

- Que linguagem o autômato M2 reconhece? (apenas mudou o estado final)

M2

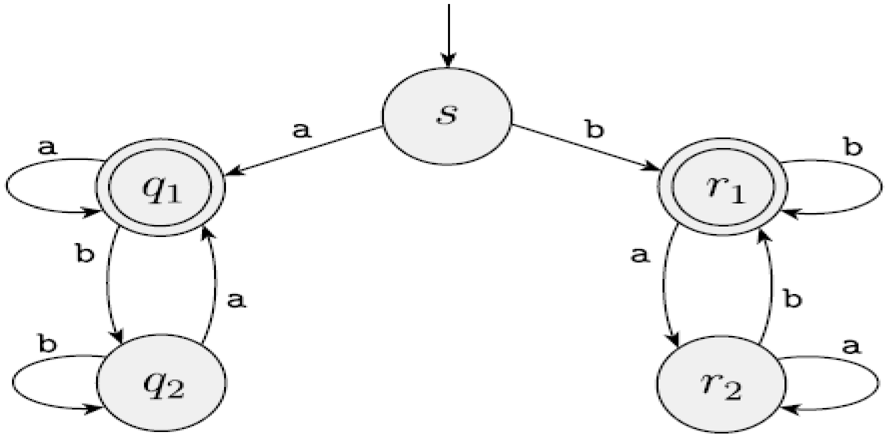


L(M2) = {w ϵ {0,1}* | w é a cadeia vazia ou termina com 0}   ou...

L(M2) = {w ϵ {0,1}* | w NÃO termina com 1}

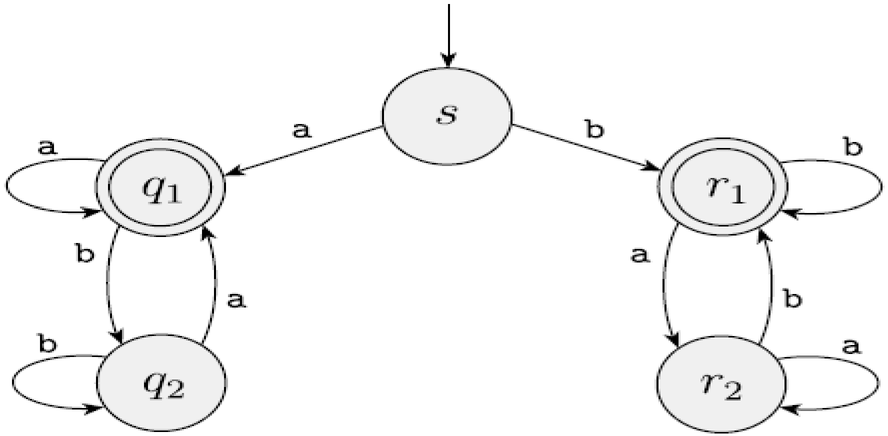
Autômatos finitos determinísticos

- Que linguagem esse autômato reconhece?



Autômatos finitos determinísticos

- Que linguagem esse autômato reconhece?

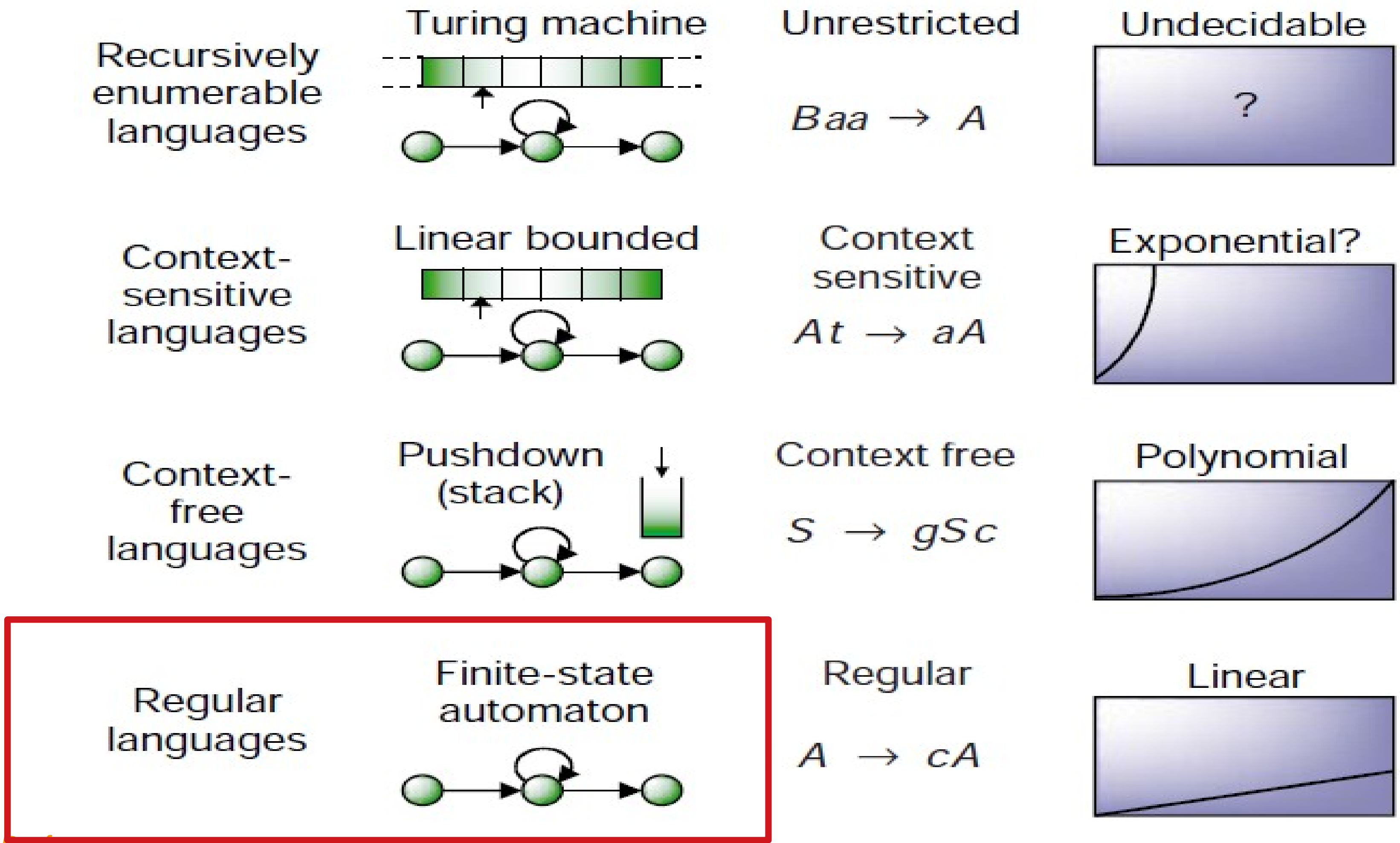


- Cadeias sobre o alfabeto {a,b} que comecem e terminem com o mesmo símbolo

Linguagem Regular

- **Definição:** Uma linguagem é chamada **linguagem regular** se algum autômato finito determinístico a reconhece

Linguagens, modelos computacionais (dispositivos, gramáticas) e suas complexidades



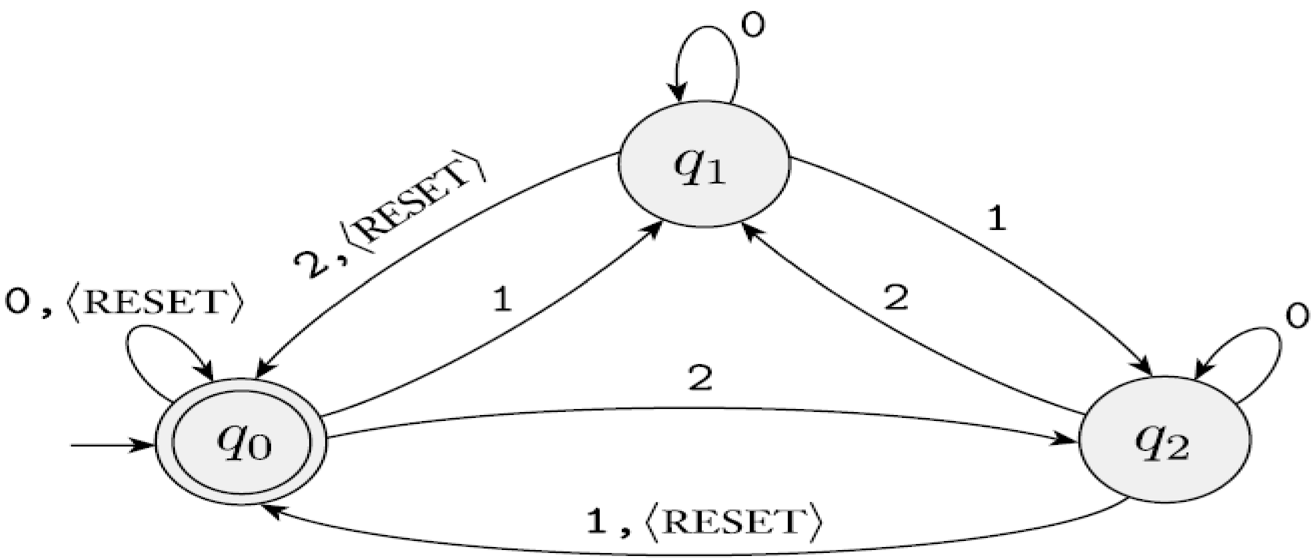
Projetando autômatos

- Pense que você é um autômato
- A cadeia de entrada pode ser arbitrariamente grande
- O número de estados é finito
- A transição se dá dados apenas o estado atual e o próximo símbolo de entrada
  - O estado atual é toda a memória que você tem
- Você recebe um símbolo por vez, e não sabe quando a cadeia vai acabar nem pode ler símbolos antecipadamente (você precisa ter sempre uma "resposta corrente" - qual a resposta se o último símbolo for este que acabei de ler?)

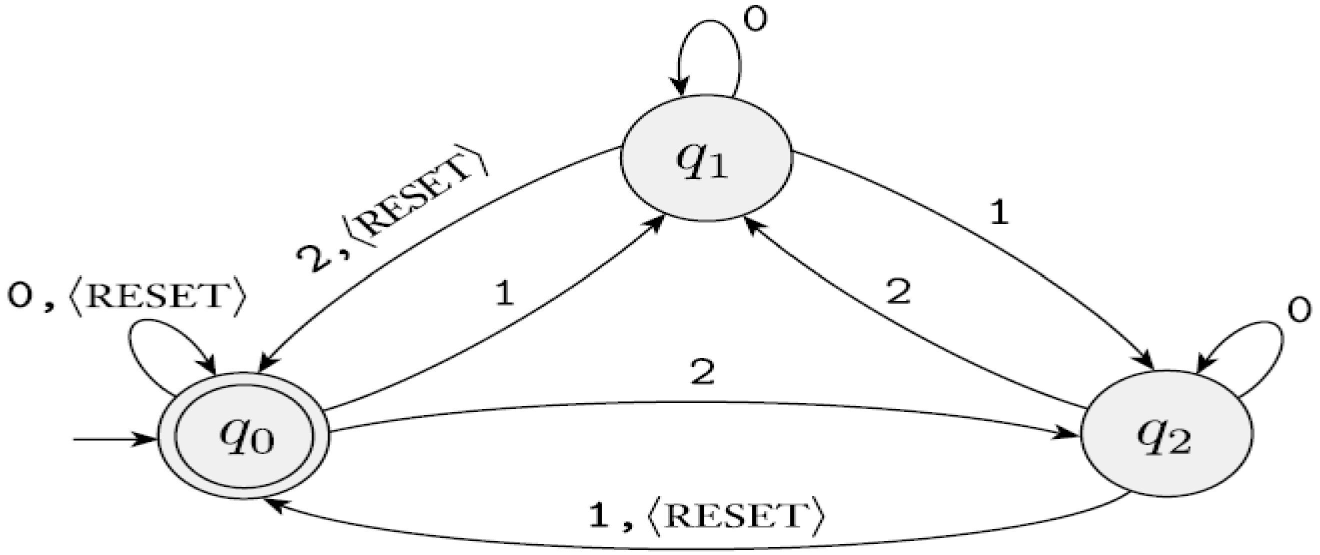
Exercício

- Projete um AFD (diagrama de estados) que, dado Σ = {0,1,2,<RESET>}, aceita a cadeia de entrada se a soma dos números for igual a 0 módulo 3 (ou seja, se a soma for um múltiplo de 3). <RESET> zera o contador. Cadeia vazia também é aceita.

Exercício - solução



Como seria um autômato que reconhece exatamente o contrário (se a soma NÃO for igual 0 módulo 3)?

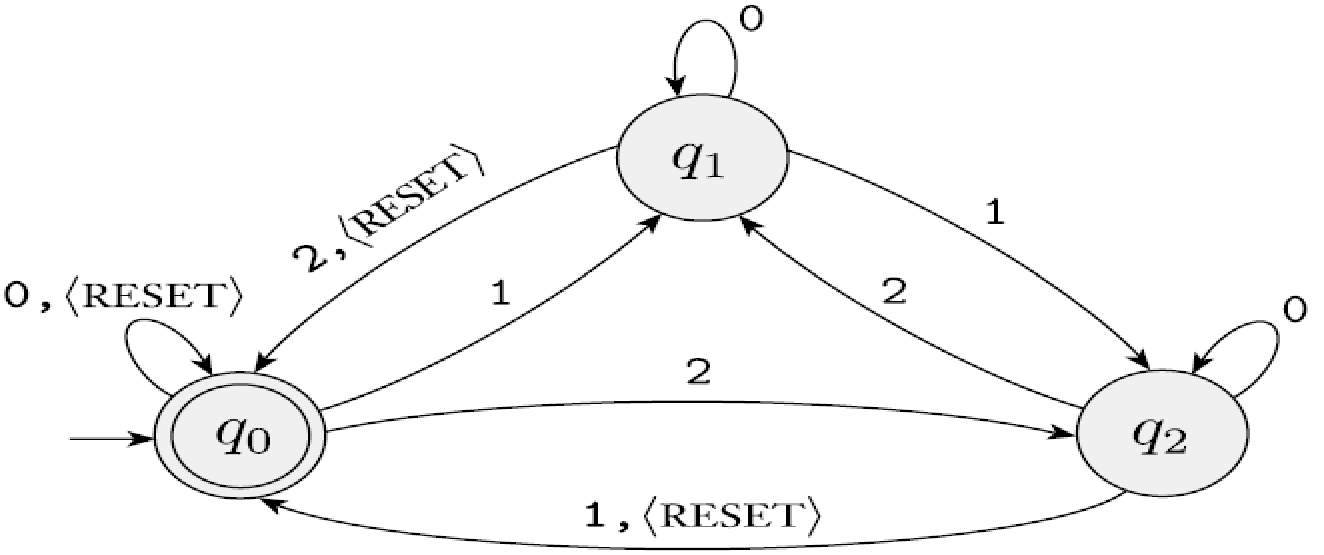


Profa. Ariane Machado Lima

Como seria um autômato que reconhece exatamente o contrário (se a soma NÃO for igual 0 módulo 3)?

Bastaria inverter a situação de cada estado como final ou não final

(Isso vale para qualquer linguagem → linguagem complementar)



Autômatos finitos

- Pode ser mais conveniente projetar o autômato usando a definição formal ao invés do diagrama de estados
- Ex: generalização do autômato anterior para aceitar somas múltiplas de i, mantendo o mesmo alfabeto (linguagem Ai)

Autômatos finitos

- Pode ser mais conveniente projetar o autômato usando a definição formal ao invés do diagrama de estados
- Ex: generalização do autômato anterior para aceitar somas múltiplas de i, mantendo o mesmo alfabeto (linguagem Ai)

autômato finito B_i, reconhecendo A_i. Descrevemos a máquina B_i formalmente da seguinte forma: $B_i = (Q_i, \Sigma, \delta_i, q_0, \{q_0\})$, onde Q_i é o conjunto de i estados $\{q_0, q_1, q_2, \ldots, q_{i-1}\}$, e desenhamos a função de transição δ_i de modo que para cada j, se B_i está em q_j, a soma corrente é j, módulo i. Para cada q_j faça

$$\delta_i(q_j, 0) = q_j,$$
$$\delta_i(q_j, 1) = q_k, \text{ onde } k = j + 1 \text{ módulo } i,$$
$$\delta_i(q_j, 2) = q_k, \text{ onde } k = j + 2 \text{ módulo } i, \text{ e}$$
$$\delta_i(q_j, \langle\text{RESET}\rangle) = q_0.$$

JFlap

Ferramenta Java para desenhar e simular autômatos, gramáticas, máquinas de Turing…

MUITO ÚTIL como recurso de estudo:

www.jflap.org

Menus File e Input

Lista MÍNIMA de exercícios do Sipser (2ª ed)

Exercícios 1.1, 1.2, 1.3, 1.5 e 1.6

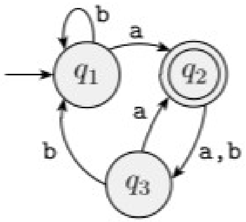
Obs: Os exercícios 1.6 que contém “ou” no enunciado da linguagem são os mais difíceis: podem tentar fazer, mas será mais fácil fazer depois das próximas aulas

Dica: Nos exercícios 1.5 (e alguns do 1.6) você pode usar o “truque” de como reconhecer a linguagem complementar

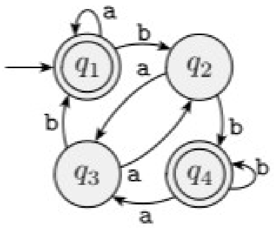
No 1.5:
* : 0 ou mais vezes
+ : 1 ou mais vezes
U: união (ou)
Ex: $(ab^+)^* \cup b^*$

Lista mínima da mínima (o que está em vermelho). Aconselhável fazer todos.

[R] **1.1** The following are the state diagrams of two AFDs, M_1 and M_2. Answer the following questions about each of these machines.



M_1



M_2

a. What is the start state?

b. What is the set of accept states?

c. What sequence of states does the machine go through on input aabb?

d. Does the machine accept the string aabb?

e. Does the machine accept the string ε?

[R] **1.2** Give the formal description of the machines M_1 and M_2 pictured in Exercise 1.1.

1.3 The formal description of a AFD M is $(\{q_1, q_2, q_3, q_4, q_5\}, \{\mathtt{u}, \mathtt{d}\}, \delta, q_3, \{q_3\})$, where δ is given by the following table. Give the state diagram of this machine.

	u	d
q_1	q_1	q_2
q_2	q_1	q_3
q_3	q_2	q_4
q_4	q_3	q_5
q_5	q_4	q_5

Difícil por enquanto… espere a aula sobre “fechamentos” se não conseguir

1.5 Each of the following languages is the complement of a simpler language. In each part, construct a AFD for the simpler language, then use it to give the state diagram of a AFD for the language given. In all parts $\Sigma = \{\mathtt{a}, \mathtt{b}\}$.

[R] **a.** $\{w \mid w$ does not contain the substring ab$\}$

[R] **b.** $\{w \mid w$ does not contain the substring baba$\}$

c. $\{w \mid w$ contains neither the substrings ab nor ba$\}$

d. $\{w \mid w$ is any string not in $\mathtt{a}^*\mathtt{b}^*\}$

e. $\{w \mid w$ is any string not in $(\mathtt{ab}^+)^*\}$

f. $\{w \mid w$ is any string not in $\mathtt{a}^* \cup \mathtt{b}^*\}$

g. $\{w \mid w$ is any string that doesn't contain exactly two a's$\}$

h. $\{w \mid w$ is any string except a and b$\}$

1.6 Give state diagrams of AFDs recognizing the following languages. In all parts the alphabet is $\{0,1\}$.

a. $\{w \mid w$ begins with a 1 and ends with a 0$\}$

b. $\{w \mid w$ contains at least three 1s$\}$

c. $\{w \mid w$ contains the substring 0101, i.e., $w = x0101y$ for some x and $y\}$

d. $\{w \mid w$ has length at least 3 and its third symbol is a 0$\}$

e. $\{w \mid w$ starts with 0 and has odd length, or starts with 1 and has even length$\}$

f. $\{w \mid w$ doesn't contain the substring 110$\}$

g. $\{w \mid$ the length of w is at most 5$\}$

h. $\{w \mid w$ is any string except 11 and 111$\}$

i. $\{w \mid$ every odd position of w is a 1$\}$

j. $\{w \mid w$ contains at least two 0s and at most one 1$\}$

k. $\{\varepsilon, 0\}$

l. $\{w \mid w$ contains an even number of 0s, or contains exactly two 1s$\}$

m. The empty set

n. All strings except the empty string