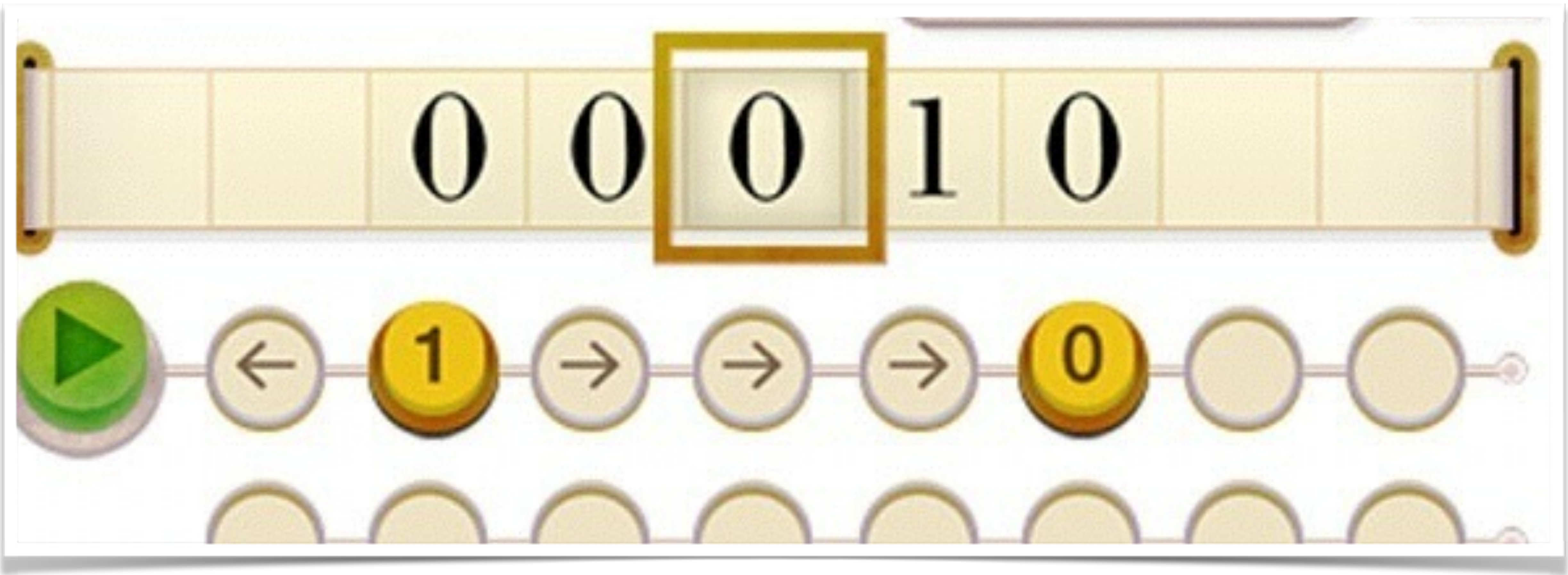
Teoria Da Computação

2014-2



Prof. Adolfo Guimarães
E-mail: adolfoguimaraes@gmail.com
Twitter: @guimaraesadolfo

Unit
UNIVERSIDADE TIRADENTES

Autômatos Finitos

Linguagens Regulares

- Podemos fazer o estudo das linguagens regulares por meio de três formalismo:
  - Expressão Regular (formalismo denotacional)
  - Gramática Regular (formalismo axiomático)
  - Autômato Finito (formalismo operacional)
    - as linguagens são representadas por máquinas abstratas ou autômatos, baseados em estados, em instruções primitivas e em como cada símbolo modifica cada estado;
    - é descrito basicamente como um sistema de estados finitos.

Sistema de Estados Finitos

- Um sistema de estado finitos é um modelo matemático de sistemas com entradas e saídas discretas.

- Podem assumir um número finito e predefinido de estados.

- Cada estado reúne somente informações necessárias para determinar as ações para a próxima entrada.

- Um forte motivacional para o estudo de sistemas de estados finitos é o fato de poderem ser associados a diversos tipos de sistemas naturais e construídos.

UNIVERSIDADE TIRADENTES

Autômato Finito

- O autômato finito é um sistema de estados finitos que permite reconhecer linguagens regulares;

- Também denominado como reconhecedor, um sistema formal capaz de aceitar todas as sentenças que pertençam a uma determinada linguagem, rejeitando todas as demais;

- Um autômato finito tem um conjunto finito de estados, e seu "controle" se desloca de estado para estado em respostas a "entradas" externas.

Autômato Finito

- Podemos classificá-los como:
  - **determinístico:** dependendo do estado corrente e do símbolo lido, o sistema pode assumir um único estado bem determinado. Em outras palavras, o autômato não pode estar em mais de um estado em um determinado instante;
  - **não determinístico:** dependendo do estado corrente e do símbolo lido, o sistema pode assumir um conjunto de estados alternativos (pode estar em mais de um estado em um determinado instante);
  - **com movimentos vazio:** dependendo do estado corrente e sem ler qualquer símbolo, o sistema pode assumir um conjunto de estados alternativos.

UNIVERSIDADE TIRADENTES

Autômato Finito Determinístico

- Um **autômato finito determinístico** pode ser visto como uma máquina constituída, basicamente, de três partes:
  - **Fita:** dispositivo de entrada que possui a informação a ser processada;
  - **Unidade de Controle:** reflete o estado corrente da máquina. Possui uma unidade de leitura a qual acessa uma célula da fita de cada vez e movimenta-se exclusivamente para a direita;
  - **Programa ou Função de transição:** função que comanda as leituras e define o estado da máquina.

Autômato Finito Determinístico

- Um autômato finito determinístico consiste em:
  - Um conjunto finito de estados, Q
  - Um conjunto finito de símbolos de entrada, ∑
  - Um função de transição, ∂
    - Dado um estado e um símbolo de entrada, retorna um estado;
    - Se **q** é um estado, e **a** é um símbolo de entrada, então **∂(q,a)** é o estado **p** tal que existe um arco identificado por **a** de **q** até **p**;
    - Um estado inicial que está contido em **Q**
    - Um conjunto de estados finais ou de aceitação **F,** sendo F um subconjunto de **Q.**

Autômato Finito Determinístico

- Podemos definir um autômato como sendo:

$$A = \{Q, \Sigma, \partial, q_0, F\}$$

Autômato Finito Determinístico

- **Exemplo 1:** Vamos especificar formalmente um AFD que aceita todas e somente as strings de 0's e 1's que têm sequência 01 em algum lugar na string.

L = {w | w é da forma x01y para alguns strings x e y que consiste somente em 0's e 1's}

ou

L = {x01y | x e y são quaisquer strings de 0's e 1's}

UNIVERSIDADE TIRADENTES

Exemplo 1

- **$A = \{\{q_0,q_1,q_2\}$, $\{0,1\}$, ∂ , q_0 , $\{q_1\}\}$**, onde **∂** é a função de transição:
  - $\partial(q_0,1) = q_0$
  - $\partial(q_0,0) = q_2$
  - $\partial(q_2,0) = q_2$
  - $\partial(q_2,1) = q_1$
  - $\partial(q_1,0) = \partial(q_1,1) = q_1$

Autômatos Finitos Determinísticos

- Como representar um autômato?
  - Diagrama de Transições (grafo)
  - Tabela de transações, onde as linhas representam os estados e as colunas as entradas.

Diagrama de Transições

- Para cada estado em **Q** existe um nó correspondente;

- Para cada estado **q** em **Q** e para cada símbolo **a** em **Σ**, seja **∂(q,a) = p**. Então, o diagrama de transições tem um arco de **q** para **p,** rotulado por **a**;

- Existem uma seta no estado inicial $\mathbf{q_0}$ identificado como **início**. Essa seta não se origina em nenhum nó;

- Os nós correspondentes aos estados de aceitação (conjunto F) são marcados por um círculo duplo. Estados que não estão em F são representados por um único círculo.

Diagrama de Transições

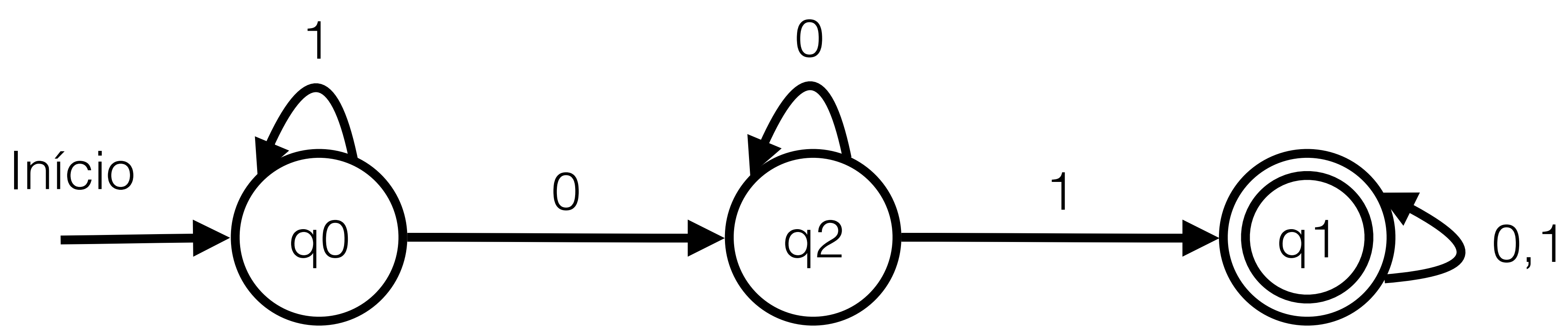


Tabela de Transições

	0	1
→ q0	q2	q0
* q1	q1	q1
q2	q2	q1

UNIVERSIDADE TIRADENTES

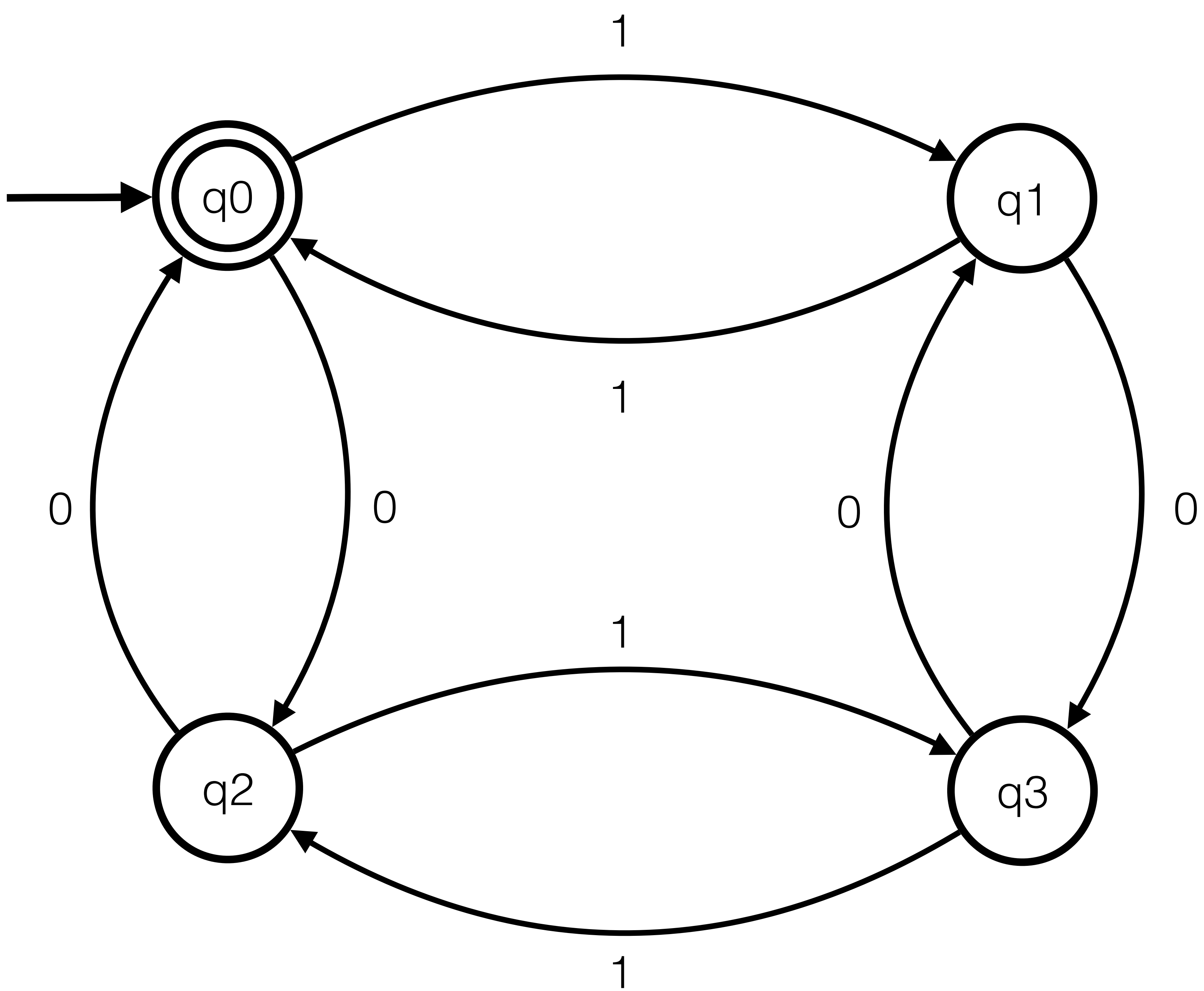
Exemplo 2

- Vamos projetar um AFD para aceitar a linguagem:
  - L = {w | w tem ao mesmo tempo um número par de 0's e um número par de 1's}

Exemplo 2

- Vamos projetar um AFD para aceitar a linguagem:
  - L = {w | w tem ao mesmo tempo um número par de 0's e um número par de 1's}
    - q0: o número de 0's e 1's são ambos pares
    - q1: o número de 0's é par e 1's é ímpar
    - q2: o número de 1's é par e 0's é ímpar
    - q3: o número de 0's e 1's são ambos ímpares

Exemplo 2



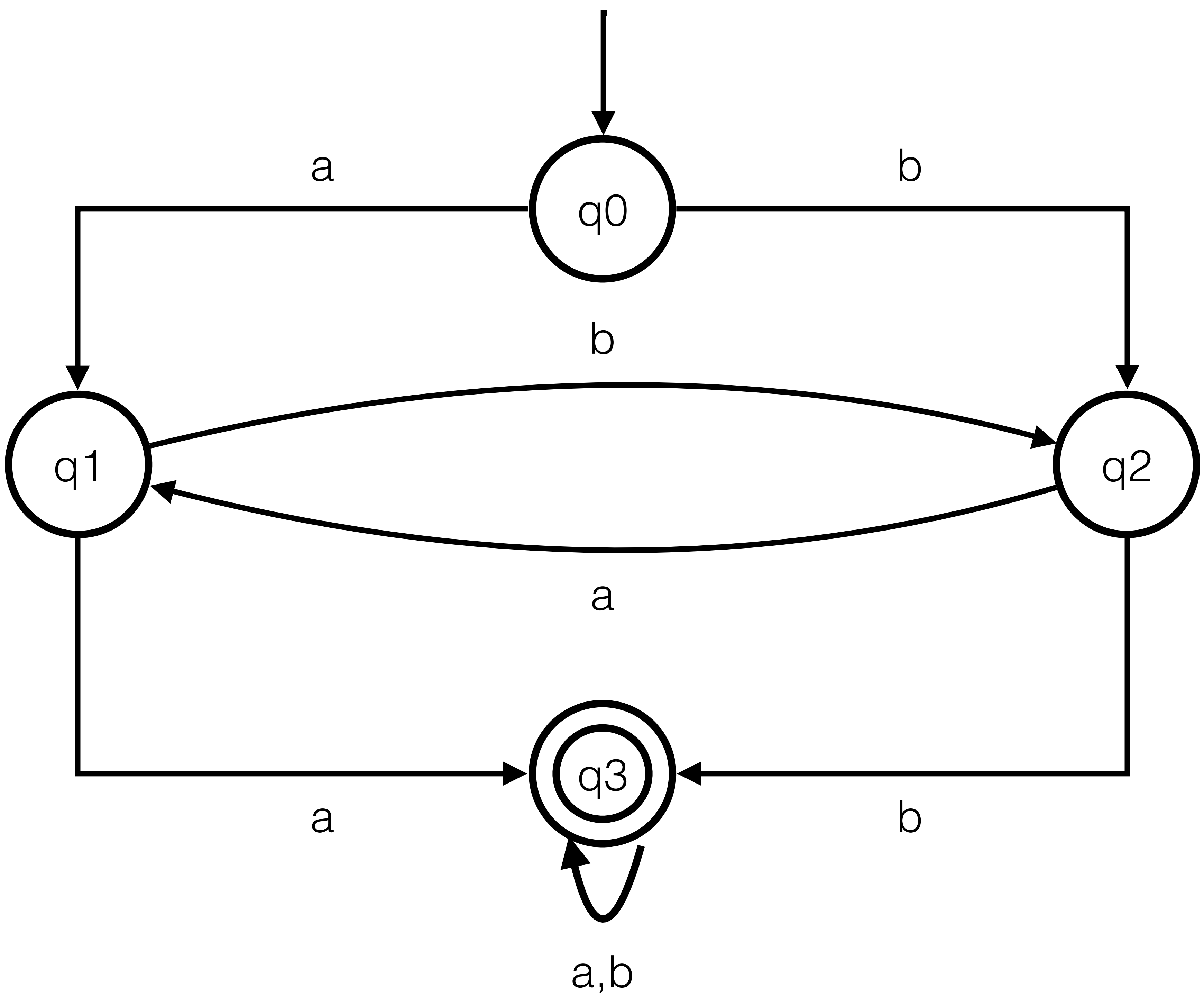
Autômatos Finitos Determinísticos

- Vale lembrar que um autômato finito sempre pára
  - Um autômato finito sempre pára ao processar qualquer entrada pois, como qualquer palavra é finita, e como um novo símbolo da entrada é lido a cada aplicação da função programa, não existe a possibilidade de ciclo (loop) infinito.
- A parada de um autômato para um entrada w pode ser de duas formas:
  - Aceita a entrada w
  - Rejeita a entrada w

Exemplo 3

- Defina um AFD para a seguinte linguagem:
  - L = {w | w possui aa ou bb como uma subpalavra}

Exemplo 3



Linguagem de um DFA

- A linguagem de um DFA $A = (Q,\Sigma,\partial,q0,F)$ é denotada por $L(A)$ e definida por:

$$L(A) = \{w \mid \partial'(q0,w) \text{ está em } F\},$$

onde ∂' é uma função de transição estendida que descreve o que acontece quando começamos em qualquer estado e seguimos qualquer sequências de entradas.

- Se L é L(A) para algum DFA A, dizemos que L é uma **linguagem regular**.

Linguagem de um DFA

- Definimos a função de transição estendida por indução sobre o comprimento do string de entrada, da seguinte forma:

  $\partial'(q,\epsilon) = q$

  $\partial'(q,w) = \partial(\partial'(q,x),a)$

  onde w é um string da forma xa, ou seja, a é o último símbolo de w, e x é o string que consiste em tudo, menos o último símbolo.

Linguagem de um DFA

- Podemos mostrar a construção da definição de função estendida (∂') para o autômato do exemplo 2. Utilize a string 110101 como entrada.

- Como esse string tem números pares de 0's e 1's, esperamos que esse string seja aceito pela linguagem. Ou seja, queremos que:

$$\partial'(q0,110101) = q0$$

pois q0 é o único estado de aceitação.

Linguagem de um DFA

- $\partial'(q0,\epsilon) = q0$
- $\partial'(q0,1) = \partial(\partial'(q0,\epsilon),1) = \partial(q0,1) = q1$
- $\partial'(q0,11) = \partial(\partial'(q0,1),1) = \partial(q1,1) = q0$
- $\partial'(q0,110) = \partial(\partial'(q0,11),0) = \partial(q0,0) = q2$
- $\partial'(q0,1101) = \partial(\partial'(q0,110),1) = \partial(q2,1) = q3$
- $\partial'(q0,11010) = \partial(\partial'(q0,1101),0) = \partial(q3,0) = q1$
- $\partial'(q0,110101) = \partial(\partial'(q0,11010),1) = \partial(q1,1) = q0$

Exercícios DFA - 01

- Defina um autômato finito que aceite strings de a's e b's que comecem com ab. Represente o autômato formalmente e através do diagrama e da tabela de transições.

Exercícios DFA - 02

- Defina um autômato finito que aceite strings de 0's e 1's terminando com 011.

Exercícios DFA - 03

- Defina um autômato finito que aceite strings de a's e b's tendo um substring aa.

UNIVERSIDADE TIRADENTES

Exercícios DFA - 04

- Defina um autômato que aceite strings de a’s e b’s exceto aqueles strings que tenham um substring aab.

Autômatos Finitos Não-Determinístico

- Um autômato finito não-determinístico tem o poder de estar em vários estados ao mesmo tempo;

- Os NFAs aceitam as linguagens regulares, da mesma maneira que fazem os DFAs;

- Frequentemente, os NFAs são mais sucintos e mais fáceis de projetar que os DFAs;

- Os DFAs podem ter exponencialmente mais estados que o NFA;

- A principal diferença está na função de transição. Em um NFA, ∂ é uma função que recebe um estado e um símbolo de entrada e retorna um conjunto de zero, um ou mais estados (em vez de retornar um estado, como um DFA deve fazer).

UNIVERSIDADE TIRADENTES

Autômatos Finitos Não-Determinístico

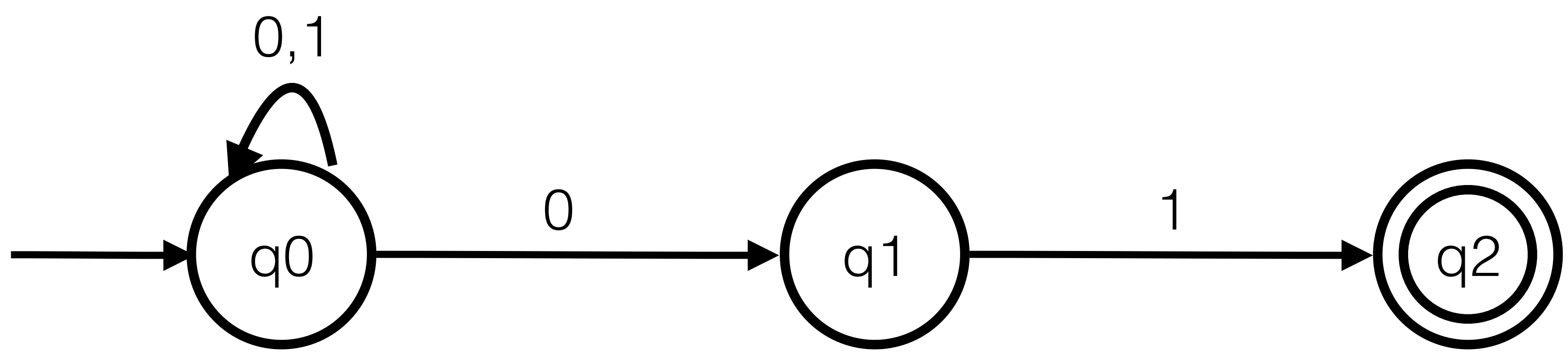
- Formalmente, podemos definir um NFA A como:
- A = {Q,∑,∂,q0,F}, onde:
  - Q é o conjunto finito de estados
  - ∑ é um conjunto finito de símbolos de entrada
  - q0, um elemento de Q, é o estado inicial
  - F, um subconjunto de Q, é o conjunto de estados finais
  - ∂ é a função de transição que recebe um estado q ∊ Q e um símbolo de entrada em ∑ como argumentos e retorna um subconjunto de Q.

Exemplo 4

- Defina uma autômato finito não determinístico cujo trabalho é aceitar todos os strings de 0’s e 1’s que terminam em 01 e somente eles.

Exemplo 4

- Defina uma autômato finito não determinístico cujo trabalho é aceitar todos os strings de 0's e 1's que terminam em 01 e somente eles.



Exemplo 4

- Formalmente, definimos o autômato como:

$$A = (\{q0,q1,q2\},\{0,1\},\partial,q0,\{q2\})$$

onde ∂ é a função de transição definida pela tabela:

	0	**1**
→ q0	{q0,q1}	{q0}
q1	∅	{q2}
***q2**	∅	∅

Exemplo 5

- Defina um autômato finito não-determinístico que reconheça strings que possuam aa ou bb como substring.

Exemplo 6

- Defina um autômato finito não-determinístico que aceite strings que possuam aaa como sufixo. Utilize o alfabeto $\sum = \{a,b\}$.

A linguagem de um NFA

- Como foi mostrado, um NFA aceita um string w se é possível tomar qualquer sequência de escolhas do próximo estado, quanto são lidos os caracteres de w, e ir do estado inicial para algum estado de aceitação.

- Formalmente, se $A = (Q,\sum,\partial,q0,F)$ é um NFA, então:

$$L(A) = \{w \mid \partial'(q0,w) \cap F \neq \varnothing\}$$

UNIVERSIDADE TIRADENTES

A linguagem de um NFA

- A função de transição estendida (∂') para um NFA também pode ser definida por um indução, onde:

$$\partial'(q,\epsilon) = \{q\}$$

$$\partial'(q,w) = \bigcup_{i=1}^{k} \partial(p_i,a)$$

A linguagem de um NFA

- Vamos usar a função estendida de um NFA para descrever o processamento da entrada 00101 pelo NFA do Exemplo 4.

- $\partial'(q0,\epsilon) = \{q0\}$
  $\partial'(q0,0) = \partial(q0,0) = \{q0,q1\}$
  $\partial'(q0,00) = \partial(q0,0) \cup \partial(q1,0) = \{q0,q1\} \cup \varnothing = \{q0,q1\}$
  $\partial'(q0,001) = \partial(q0,1) \cup \partial(q1,1) = \{q0\} \cup \{q2\} = \{q0,q2\}$
  $\partial'(q0,0010) = \partial(q0,0) \cup \partial(q2,0) = \{q0,q1\} \cup \varnothing = \{q0,q1\}$
  $\partial'(q0,00101) = \partial(q0,1) \cup \partial(q1,1) = \{q0\} \cup \{q2\} = \{q0,q2\}$

Exercício NFA - 01

- Defina um NFA que aceite a seguinte linguagem $L = \{w \mid w \in abab^n \text{ ou } aba^n \text{ onde } n \geq 0\}$

Exercício NFA - 02

- Projete um NFA que reconheça o conjunto de string abc, abd e aacd. Suponha que o alfabeto seja {a, b, c, d}.

UNIVERSIDADE TIRADENTES

Exercício NFA - 03

- Defina um NFA para uma linguagem formada pelo conjunto de strings sobre o alfabeto {0,1,…,9} tal que o dígito final tenha aparecido antes

Referência

- HOPCROFT, J., MOTWANI, R., ULLMAN, J. Introduction to Automata Theory, Languages and Computation. Addison Wesley; 3 edition, 2006.

UNIVERSIDADE TIRADENTES

Dúvidas

Unit
UNIVERSIDADE TIRADENTES