

Autômatos Finitos

Introdução

- Autômatos finitos
 - Usos comuns:
 - Projetar e verificar circuitos digitais
 - Analisador léxico de compiladores
 - Divide o texto em unidades lógicas (ids, palavras-chave, pontuação, etc)
 - Software detector de padrões em grandes corpos de texto
 - Verificador de sistemas
 - Número finito de estados distintos

Autômatos Finitos

Definição: Autômato Finito Determinístico (AFD)

Um *autômato finito determinístico* (AFD), ou simplesmente *autômato finito* M é uma quintupla:

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

onde:

- Σ - Alfabeto de símbolos de entrada
- Q - Conjunto finito de estados possíveis do autômato
- δ - Função programa ou função de transição $\delta: Q \times \Sigma \rightarrow Q$
- q_0 - Estado inicial tal que $q_0 \in Q$
- F - Conjunto de estados finais, tais que $F \subseteq Q$.

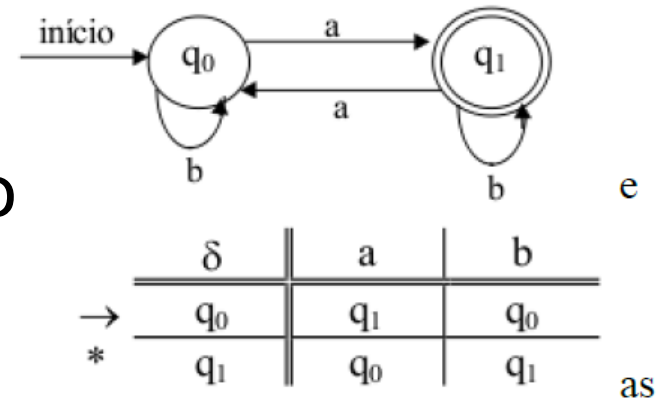
Representação

- Diagramas de transição

- grafo direcionado e rotulado

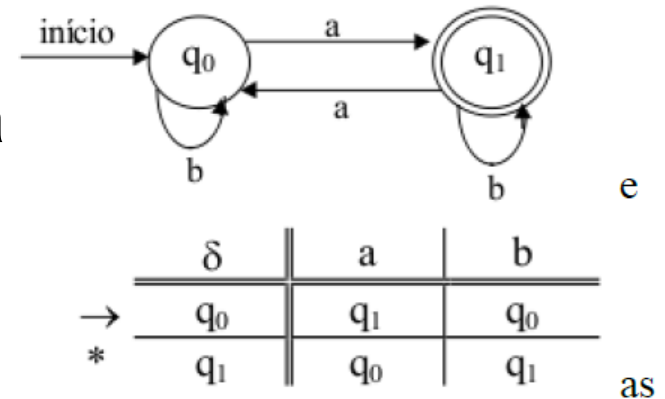
- Os vértices representam os estados

- fisicamente são representados por círculos
 - Estado inicial possui uma seta
 - com rótulo “início”
 - Os estados finais possuem círculos duplos



Autômato Finito Determinístico

- Reconhecer sentenças
 - Para reconhecer a sentença “abbaba” o autômato realiza as seguintes transições, partindo do estado inicial: “q0, q1, q1, q1, q0, q0, q1”

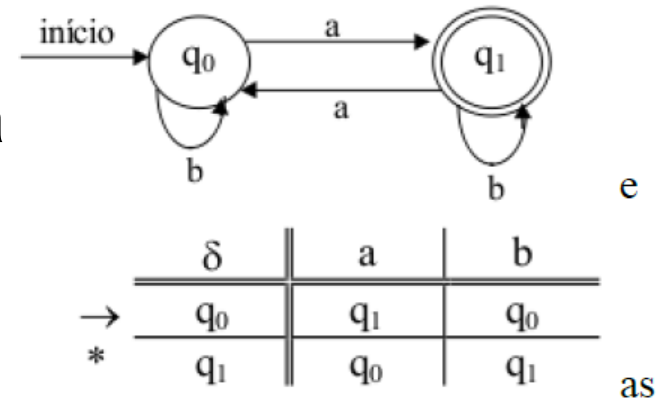


- Como terminou em um estado final, a sentença faz parte da linguagem definida.

Autômato Finito Determinístico

- Reconhecer sentenças

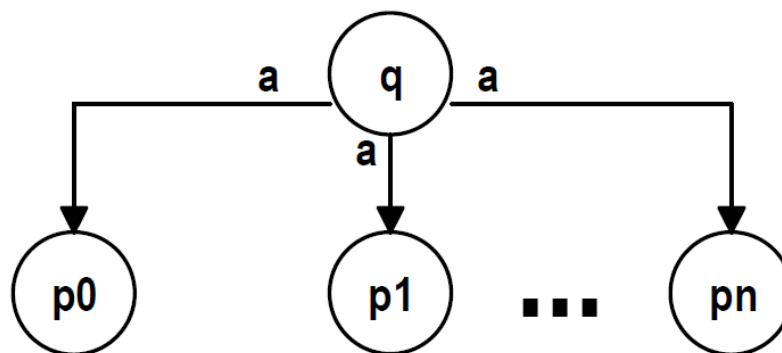
- Para reconhecer a sentença “ababaab” o autômato realiza as seguintes transições, partindo do estado inicial: “q0, q1, q0, q0, q1, q0, q0”



- Como terminou em um estado não-final, a sentença não faz parte da linguagem definida.

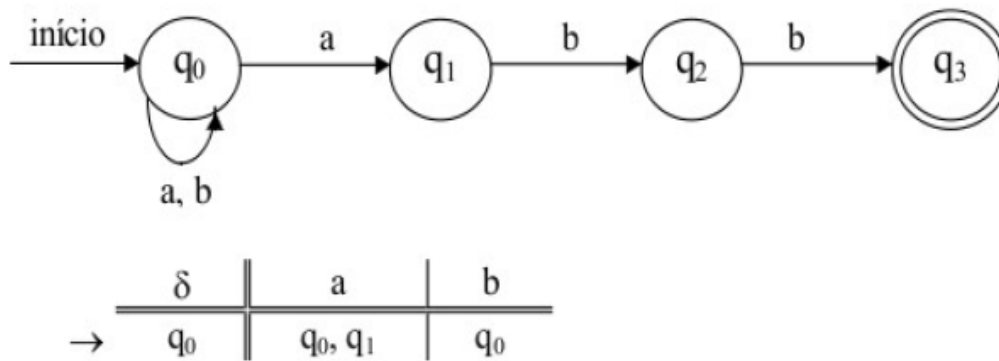
Autômato Finito não-Determinístico

- A partir do estado “q”, com o símbolo “a” na entrada pode ir tanto para o estado p1, como para o estado p2, ..., como para o estado pn.



Autômato Finito não-Determinístico

Dado o AFN a seguir e a sentença: “ababb”



Há três possibilidades no reconhecimento da sentença:

- um caminho termina num estado não-final (q_0);
- um caminho chega a um estado (q_2) onde não há transição definida para o próximo símbolo (a);
- um caminho chega a um estado final (q_3) no final da sentença.

Como há a possibilidade de chegar a um estado final com a sentença, então a mesma é válida, ou seja, deve ser reconhecida como parte da linguagem definida.

Conceitos centrais

- Alfabeto
 - Conjunto de símbolos (finito e não-vazio)
 - $\Sigma = \{0,1\}$, o alfabeto binário
 - $\Sigma = \{a,b,c,\dots,z\}$, conjunto de letras minúsculas
 - Conjunto de caracteres ASCII
- String
 - Palavra ou cadeia
 - Seqüência finita de símbolos do alfabeto

Conceitos centrais

- Linguagem
 - Conjunto de strings escolhidos a partir de um alfabeto base.
 - Exemplos: Português, linguagem c, etc.
- Problema
 - Decidir se um dado string é elemento de uma linguagem específica.

Gramáticas

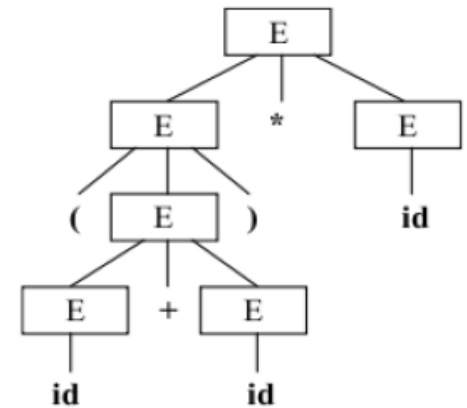
- Gramáticas Livre de Contexto
 - Descreve uma linguagem por regras recursivas (produções)
 - Cada produções consiste em:
 - Variável de cabeça
 - Um corpo, que pode ser:
 - Um string de zero ou mais variáveis
 - Terminais

Derivações

Árvore de Derivação

$$\begin{array}{l} E ::= E + E \\ \quad | E * E \\ \quad | (E) \\ \quad | id \end{array}$$

“(id+id)*id”



Se cada passo na produção de uma derivação é aplicado na variável mais a esquerda, então a derivação é chamada derivação mais a esquerda.

Similarmente, uma derivação onde a cada passo a variável mais a direita é substituída, é chamada de derivação mais a direita.

Derivações

- Ambigüidade:
 - Se uma gramática G tem mais de uma árvore de derivação para uma mesma sentença, e um mesmo método de derivação (mais a direita ou mais a esquerda) então G é chamada de gramática ambígua.