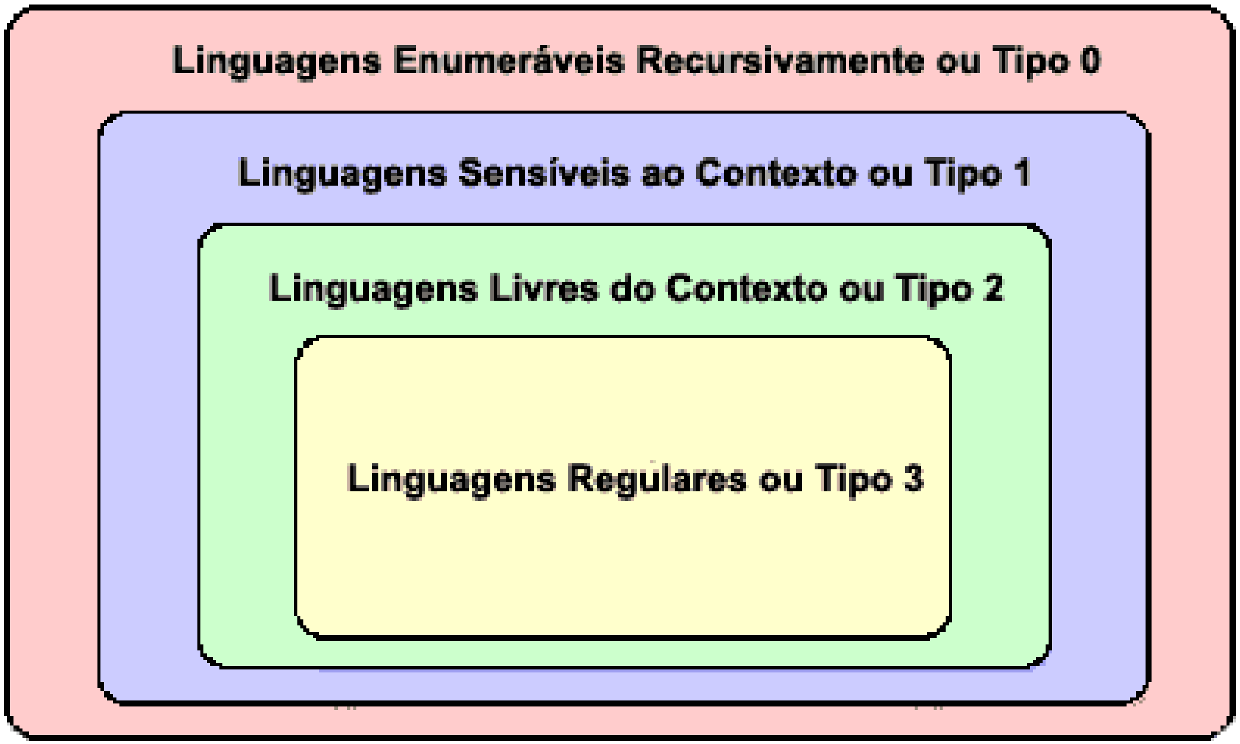
Linguagens Regulares

Prof. Daniel Oliveira

Linguagens Regulares

- Linguagens Regulares ou Tipo 3



Hierarquia de Chomsky

Linguagens Regulares

- Aborda-se os seguintes formalismos:
  - Autômatos Finitos
  - Expressões Regulares
  - Gramática Regular

Linguagens Regulares

- Classe de linguagens mais simples
- Utilização de algoritmos de reconhecimento de pouca complexidade
- Não suporta duplo balanceamento:
  - Por Exemplo: Parênteses balanceados – para cada parênteses aberto deve haver um fechando. Comum na maioria das linguagens de programação; C,C++,Pascal...
  - Ou seja, a maioria das linguagens de programação são não regulares.

Sistema de Estados Finitos

- Modelo matemático de sistema com entradas e saídas discretas
- Assumi-se um número finito e predefinido de estados
- Cada estado mantém apenas as informações do passado necessárias para determinar as ações para a próxima entrada.
- Como todos os estados possíveis são conhecidos, os mesmos podem ser definidos de partida.

Sistema de Estados Finitos

- Exemplo de um sistema de Estados Finitos:
  - Elevador – Sistema que não memoriza as requisições anteriores. Cada “estado” armazena as informações “andar corrente” e “direção de movimento”.

- Quando um sistema tem uma quantidade tão elevada de estados que esta abordagem é pouco prática diz-se que ocorre uma *explosão de estados.*
  - Por exemplo: cérebro humano tem aproximada 2^{35} neurônios, o elevado número de combinação de estados possíveis ocasiona uma explosão de estados.

Sistema de Estados Finitos

- Sistemas complexos são construídos a partir de partes mais elementares e menos complexas.
- Três formas de composição:
  - *Seqüencial* – A execução da próxima componente depende da componente anterior.
  - *Concorrente* – Basea-se em componente independentes, podendo ser processadas em paralelo.
  - *Não-determinístico* – A próxima componente a ser executada é uma escolha entre diversas componentes alternativas

Autômato Finito

- Sistema de estado finitos
- Modelo computacional do tipo seqüencial
- Formalismo operacional ou reconhecedor, o qual pode ser:
  - *Determinístico* – dependendo do estado corrente e do símbolo lido, o sistema pode assumir um único estado bem determinado.
  - *Não determinístico* – dependendo do estado corrente e do símbolo lido, o sistema pode assumir um conjunto de estados alternativos.
  - *Com movimentos vazios* – dependendo do estado atual e sem ler qualquer símbolo, o sistema pode assumir um conjunto de estado (portanto é não determinístico)

Autômato Finito

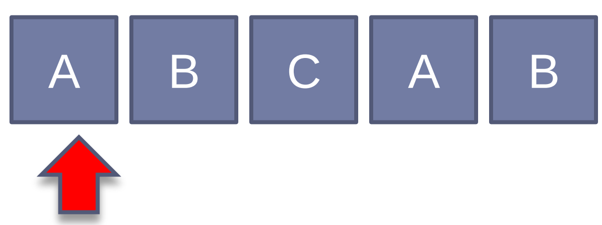
- Modelo matemático com entradas e saídas.

- Se é fornecido ao AF uma seqüência de símbolos como entrada, ele responderá se esta seqüência pertence a linguagem ou não;

- Um estado resume os estados anteriores pelos quais passou e os símbolos que já foram lidos na entrada.

Autômato Finito

- Um autômato finito determinístico (AFD), pode ser visto como sendo uma máquina com três partes:
  - Fita – Dispositivo de entrada que contém a informação a ser processada
  - Unidade de controle – reflete o estado da máquina. Possui uma unidade de leitura a qual acessa uma célula da fita de cada vez e movimenta-se para a direita
  - Programa, Função programa ou Função de Transição – Função que comanda as leituras e define o estado da máquina.



Autômato Finito Determinístico

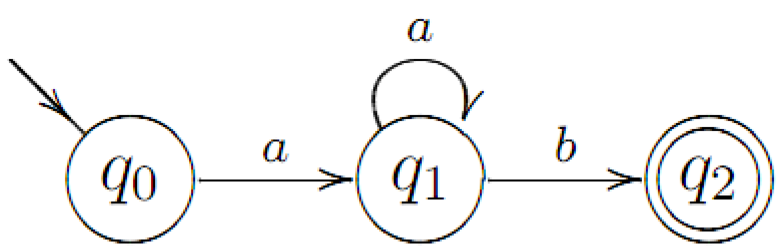
- É uma 5-upla ordenada: $M = (\Sigma, Q, \delta, q_0, F)$

  - Σ – é o alfabeto de símbolos de entrada ou alfabeto de entrada
  - Q – é um conjunto de estados possíveis do autômato qual é finito
  - δ – é uma função programa ou simplesmente programa ou *função de transição*. Sendo definida para um estado p e um símbolo a , resultado no estado q: $\delta(p,a) = q$, é uma transição de autômato.
  - q_0 – Estado inicial
  - F – subconjunto de estados finais

Autômato Finito Determinístico

- Diagrama de Transição
  - Outra maneira de representar um autômato finito
  - Grafo direcionado e rotulado
  - Os vértices representam os estados,desenhados como círculos
  - As arestas representam as transições entre dois estados, sendo o rótulo o símbolo reconhecido na entrada
  - O estado inicial é marcado com uma seta
  - Os estados finais são indicados por círculos duplos.

Autômato Finito Determinístico



$M = (\{q_0, q_1, q_2\}, \{a, b\}, \delta, q_0, \{q_2\})$, e $\delta(q_0, a) = q_1$, $\delta(q_1, a) = q_1$, $\delta(q_1, b) = q_2$. Logo, $T(M) = \{a^n b | n \geq 1\}$

Autômato Finito Determinístico

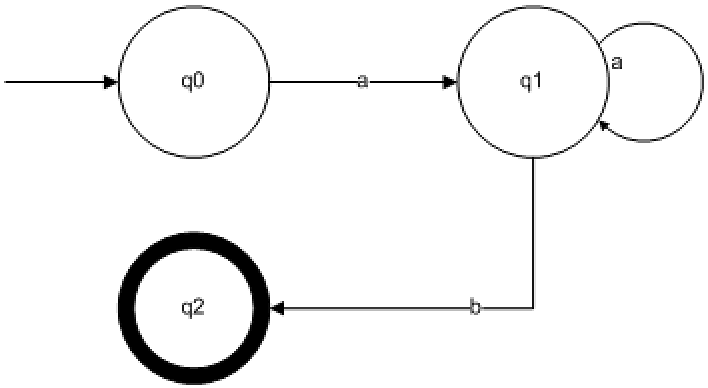
- Tabela de Transição de Estados
  - Uma terceira forma de representação de um AF
  - Tabela indicando na vertical os estados,e na horizontal os símbolos do alfabeto.No cruzamento estão as transições possíveis
  - O estado inicial é indicado por uma seta,e os estados finais por asteriscos.

Autômato Finito Determinístico

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

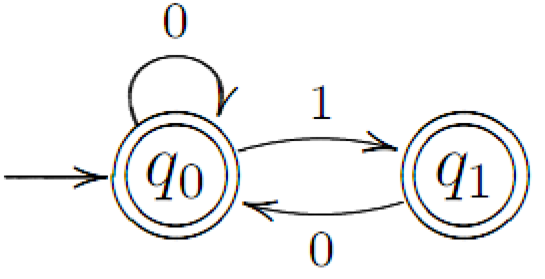
Qual seria a descrição formal do AF?
Qual seria a descrição gráfica do AF?

$M = (\Sigma, Q, \delta, q_0, F)$
$Q = \{q0, q1, q2\}$
$\Sigma = \{a, b\}$
$F = \{q2\}$
$\delta(q0,a) = q1$
$\delta(q1,a) = q1$
$\delta(q1,b) = q2$



Autômato Finito Determinístico

- AF que reconhece sentenças em {0,1}* as quais não contém dois ou mais 1's consecutivos



$$M = (K, \Sigma, \delta, q_0, F),\ K = \{q_0, q_1\},\ \Sigma = \{0, 1\},$$
$$F = \{q_0, q_1\},\ \text{com } \delta(q_0, 0) = q_0,\ \delta(q_0, 1) = q_1,\ \delta(q_1, 0) = q_0$$

Qual é a tabela de transição ?

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

Autômato Finito Determinístico

- Exemplo: aa ou bb como sub-palavra

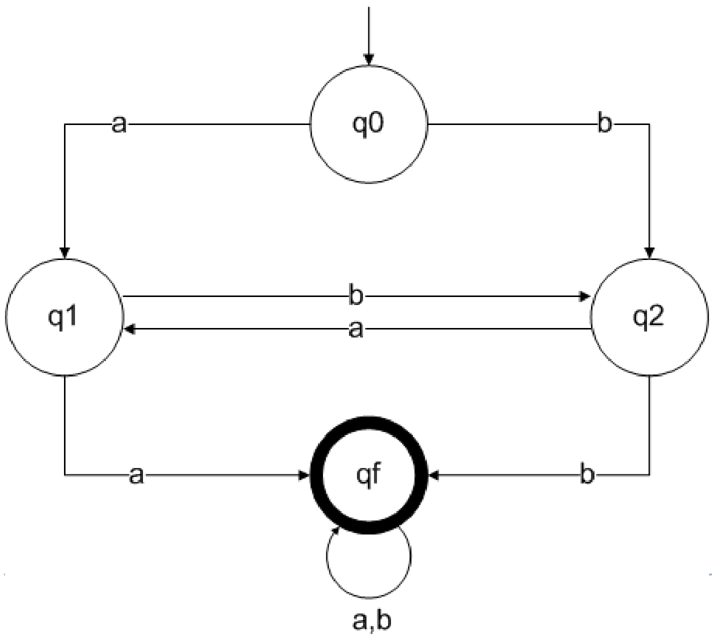
Considerando a linguagem sobre o alfabeto {a,b}

L1 = {w | w possui aa ou bb como sub-palavra}

AFD: M1 = ({a,b},{q0,q1,q2,qf},δ1,q0,{qf})

M1 para a entrada w = abba

δ1	a	b
q0	q1	q2
q1	qf	q2
q2	q1	qf
q3	qf	qf



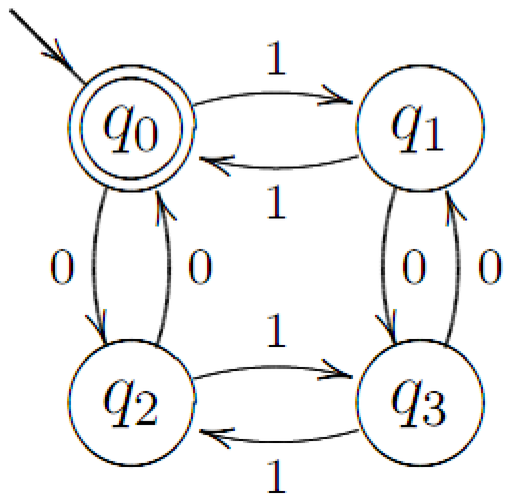
Autômato Finito Determinístico

- A linguagem aceita por um autômato finito M = (Σ, Q, δ, q0, F) denotada por:
  - ACEITA(M), ou L(M), é o conjunto de todas as palavras pertencentes a Σ* que são aceitas por M, ou seja: ACEITA(M) - {w | δ(q0, w) ∈ F}.
  - Analogamente, REJEITA(M) é o conjunto de todas as palavras pertencentes a Σ* que são rejeitadas por M.
- As seguintes afirmações são verdadeiras
  - A intersecção dos conjuntos ACEITA(M) e REJEITA(M) é vazio.
  - A união dos conjuntos ACEITA(M) e REJEITA(M) é Σ*.
  - REJEITA(M) é o complemento de ACEITA(M) em Σ*
  - ACEITA(M) é o complemento de REJEITA(M) em Σ*

Autômato Finito Determinístico

Exemplo: $M = (K, \Sigma, \delta, q_0, F)$, $\Sigma = \{0, 1\}$,
$K = \{q_0, q_1, q_2, q_3\}$, $F = \{q_0\}$.

$$
\begin{aligned}
\delta(q_0, 0) &= q_2 \\
\delta(q_1, 0) &= q_3 \\
\delta(q_2, 0) &= q_0 \\
\delta(q_3, 0) &= q_1 \\
\delta(q_0, 1) &= q_1 \\
\delta(q_1, 1) &= q_0 \\
\delta(q_2, 1) &= q_3 \\
\delta(q_3, 1) &= q_2
\end{aligned}
$$



δ	0	1
$\rightarrow *q_0$	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2

Autômato Finito Determinístico

- Qual seqüência 110101 ou 1011 pertence a L(M) ?

$(q_0,110101) \rightarrow (q_1,10101) \rightarrow (q_0,0101) \rightarrow (q_2,101) \rightarrow (q_3,01) \rightarrow (q_1,1) \rightarrow (q_0,\varepsilon)$

Assim, 110101 $\in$ L(M)

$(q_0,1011) \rightarrow (q_1,011) \rightarrow (q_3,11) \rightarrow (q_2,11) \rightarrow (q_3,\varepsilon)$

Assim, 1011 não pertence a L(M)

$L(M) = \{w \mid w \in \{0,1\}^*$ e w contém um número par de 0's e 1's$\}$

δ	0	1
$\rightarrow *q_0$	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2

Autômato Finito Não Determinístico

- Autômato Finito não Determinístico ou AFN

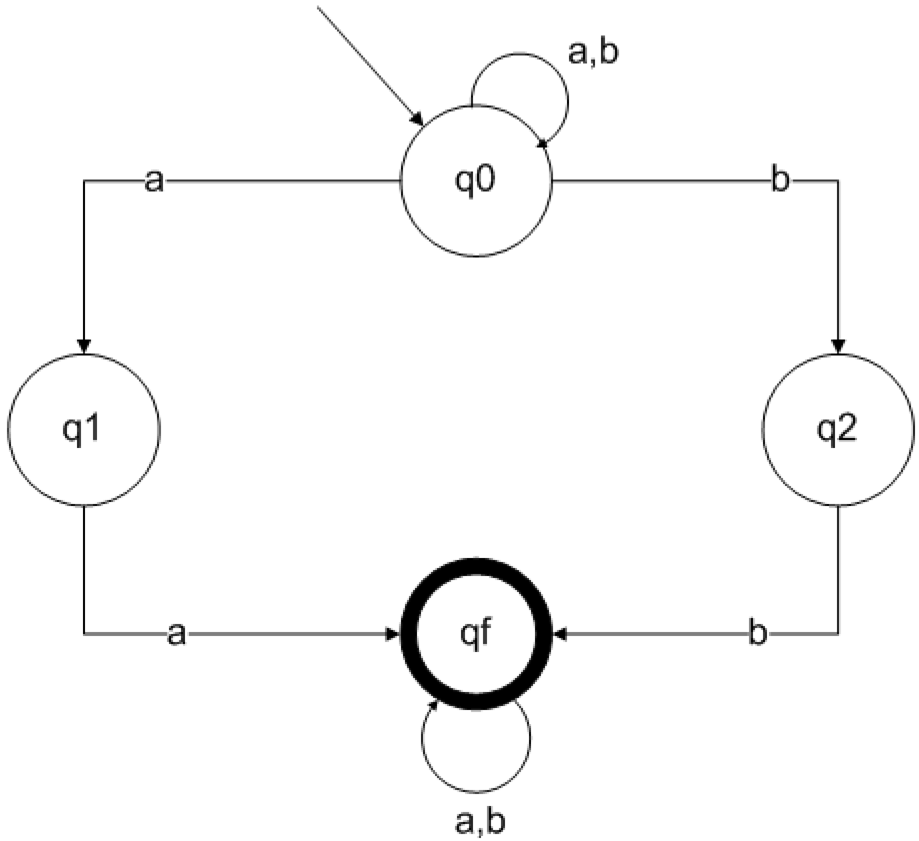
É uma 5-upla ordenada: $M = (\Sigma, Q, \delta, q_0, F)$

- Σ – é o alfabeto de símbolos de entrada ou alfabeto de entrada
- Q – é um conjunto de estados possíveis do autômato qual é finito
- δ – é uma função programa ou simplesmente programa ou *função de transição*. Sendo definida para um estado p e um símbolo a , resultado no estado q: $\delta(p,a) = \{q1,q2,...,qn\}$ é uma transição de autômato.
- q_0 – Estado inicial
- F – subconjunto de estados finais

Autômato Finito Não Determinístico

- Exemplo: aa ou bb como sub-palavra

δ	a	b
q0	{q0,q1}	{q0,q2}
q1	{qf}	-
q2	-	{qf}
q3	{qf}	{qf}



Expressões Regulares

- Toda linguagem regular pode ser descrita por uma expressão denominada de Expressão Regular
- É definida a partir de conjuntos (linguagens) básicos e operações de concatenação e de união.

Expressão Regular	Linguagem Gerada
aa	somente a palavra aa
ba*	todas as palavras que iniciam com b, seguido por zero ou mais a
(a+b)*	todas as palavras sobre {a,b}
(a+b)*aa(a+b)*	todas as palavras contendo aa como subpalavra
a*ba*ba*	todas as palavras contendo exatamente dois b
(a+b)*(aa+bb)	todas as palavras que terminam com aa ou bb
(a+ε)(b+ba)*	todas as palavras que não possuem dois a consecutivos

Expressões Regulares

- Teoremas:
  - Se r é uma expressão regular, então GERA(r) é uma linguagem regular. Aonde GERA(r) ou L(r) é a linguagem gerada por r
  - Se L é uma linguagem regular, então existe uma expressão regular r tal que: Gera(r) = L

Gramática Regular

- Definições

  - *Gramática Linear à Direita (GLD)* – todas as regras de produção são da forma: $A \rightarrow wB$ ou $A \rightarrow w$
  - *Gramática Linear à Esquerda (GLE)* – todas as regras de produção são da forma: $A \rightarrow Bw$ ou $A \rightarrow w$
  - *Gramática Linear Unitária a Direita (GLUD)* – todas as regras de produção da GLD e, com: $|w| \leq 1$
  - *Gramática Linear Unitária a Esquerda (GLUE)* – todas as regras de produção da GLE e, com: $|w| \leq 1$

Gramática Regular

- Uma gramática G é dita uma Gramática Regular (GR) se G é uma gramática linear
- Exemplo: GR $(a+b)^*(aa+bb)$, é gerada pelas seguintes gramáticas regulares:

a) *Linear á Direita*, G=({S,A},{a,b},P,S), e P possui a seguinte Produção:
$S \rightarrow aS|bS|A$
$A \rightarrow aa|bb$

b) *Linear á Esquerda*, G=({S,A},{a,b},P,S), e P possui a seguinte Produção:
$S \rightarrow Aaa|Abb$
$A \rightarrow Aa|Ab|\varepsilon$

Gramática Regular

- Teoremas

  - Se L é uma linguagem gerada por uma gramática regular, então L é uma linguagem regular
  - Se L é uma linguagem regular então existe G, gramática regular que gera L

Propriedades das Linguagens Regulares

- Se uma linguagem é regular, então é aceita por um autômato finito determinístico o qual possui um número finito e predefinido de n estados.
- A classe de Linguagens Regulares é fechada para as seguintes operações:
  - União
  - Concatenação
  - Complemento
  - Intersecção

Autômato Finito com Saída

- O conceito básico de autômato finito possui aplicações práticas restritas, pois a informação de saída é limitada à lógica binária aceita/rejeita.

- As saídas podem ser associadas:
  - às transições (Máquina de Mealy)
  - aos estados (Máquina de Moore)

Autômato Finito com Saída

- Em ambas as máquinas, a saída não pode ser lida, ou seja, não pode ser usada como memória auxiliar, e é como se segue:
  - É definida sobre um alfabeto especial, denominado alfabeto de saída, o qual pode ser igual ao alfabeto de entrada
  - A saída é armazenada em uma fita de saída independente da de entrada
  - A cabeça da fita de saída move uma célula para a direita a cada símbolo gravado.
  - O resultado do processamento do autômato finito é o seu estado final (condição aceita/rejeita) e a informação contida na fita de saída.

Máquina de Mealy

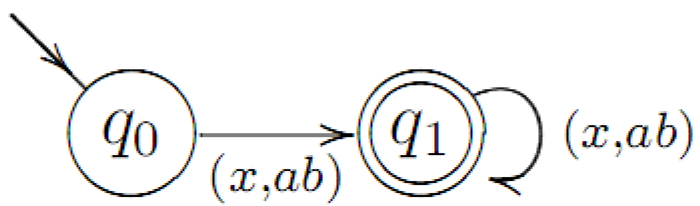
- É um autômato finito modificado de forma a gerar uma palavra de saída (a qual pode ser vazia) para cada transição da máquina.
- É uma 6-upla ordenada: M = (Σ,Q,δ,q0,F,Δ)
  - Σ – é o alfabeto de símbolos de entrada ou alfabeto de entrada
  - Q – é um conjunto de estados possíveis do autômato qual é finito
  - δ – é uma função programa ou simplesmente programa ou *função de transição*. A qual é parcial
  - q_0 – Estado inicial
  - F – subconjunto de estados finais
  - Δ – alfabeto de símbolos de saída ou alfabeto de saída

Máquina de Mealy

- A computação da máquina de Mealy para uma palavra de entrada w:
  - Aplicação sucessiva da função programa para cada símbolo de w (da esquerda para direita) até ocorrer uma condição de parada.
  - A palavra vazia como saída do autômato indica que nenhuma gravação é realizada e a cabeça da fita de saída não se move.
  - Se todas as transições geram saída vazia, então a Máquina de Mealy processa como se fosse um autômato finito.

Máquina de Mealy

- Seja $\Sigma = \{x\}$ e $\Delta = \{a,b\}$. M é uma máquina de Mealy que reconhece uma palavra w, $|w| \geq 1$, tal que $w = x^+$ e produz uma seqüência $v = (ab)^+$ em que $|v|=2|w|$.



Observação: arestas são rotuladas com um par (x, y), onde x é o símbolo lido e y é a string produzida na saída.

Máquina de Moore

- Possui uma segunda função, que gera uma palavra de saída (a qual pode ser vazia) para cada estado da máquina.
- É uma 7-upla ordenada: $M = (\Sigma, Q, \delta, q0, F, \Delta, \delta_s)$
  - Σ – é o alfabeto de símbolos de entrada ou alfabeto de entrada
  - Q – é um conjunto de estados possíveis do autômato qual é finito
  - δ – é uma função programa ou simplesmente programa ou *função de transição*. A qual é parcial
  - q_0 – Estado inicial
  - F – subconjunto de estados finais
  - Δ – alfabeto de símbolos de saída ou alfabeto de saída
  - δ_s – função de saída: $\delta_s : Q \rightarrow \Delta^*$, a qual é uma função total

Máquina de Moore

- Computação da máquina de Moore:
  - Sucessiva aplicação da função de programa para cada símbolo de w (da esquerda para direita) até ocorrer uma condição de parada, juntamente com a sucessiva aplicação da função de saída a cada estado atingido.
  - A palavra vazia como resultado da função de saída indica que nenhuma gravação é realizada e, obviamente, não se move a cabeça da fita de saída.
  - Se todos os estados geram saídas vazias então a máquina de Moore comporta-se como um AF.

Máquina de Moore

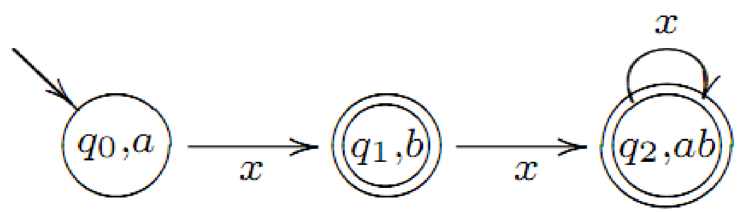
- Exemplo comum é um analisador léxico:
  - Um estado final é associado a cada unidade Léxica
  - Cada estado final possui uma saída (definida pela Função de Saída) que descreve ou codifica a unidade léxica identificada
  - Para os demais estados (não-finais), em geral, a saída gerada é a palavra vazia. Podendo ser não vazia, caso alguma informação adicional seja necessária para a codificação da unidade léxica

Máquina de Moore

- Seja Σ = {x} e Δ = {a,b}. M é uma máquina de Mealy que reconhece uma palavra w, |w| ≥ 1, tal que $w = x^+$ e produz uma seqüência $v = (ab)^+$ em que $|v|=2|w|$.

$(xxx \cdots x \Rightarrow ababab \cdots ab)$.

Solução:



Observação: os vértices são rotulados com um par (x, y), onde x é o nome do estado e y é a saída produzida pelo estado.