



UNIVERSIDAD DE CÓRDOBA
 ESCUELA POLITÉCNICA SUPERIOR
 DEPARTAMENTO DE INFORMÁTICA Y ANÁLISIS NUMÉRICO
 INGENIERÍA TÉCNICA EN INFORMÁTICA DE SISTEMAS
 SEGUNDO CURSO, SEGUNDO CUATRIMESTRE



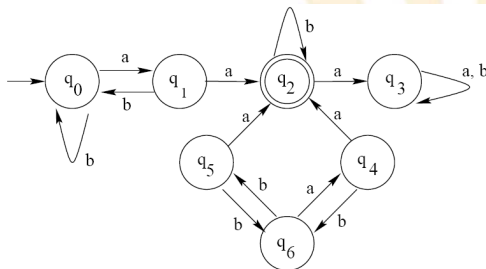
TEORÍA DE AUTÓMATAS Y LENGUAJES FORMALES

Tema 6.- Autómatas finitos
 y máquinas secuenciales

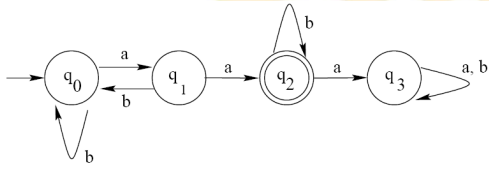


Algoritmo para la obtención de los estados accesibles

- [1] inicio
- [2] $Accesibles \leftarrow \{q_0\}$ y q_0 no marcado
- [3] mientras haya un estado no marcado $q \in Accesibles$ hacer
- [4] Marcar el estado q
- [5] para cada $\sigma \in \Sigma$ hacer
- [6] si $\delta(q, \sigma) = q' \notin Accesibles$
- [7] entonces $Accesibles \leftarrow Accesibles \cup \{q'\}$
- [8] y q' no marcado
- [9] fin si
- [10] fin para
- [11] fin mientras
- [12] fin



Autómata con estados inútiles



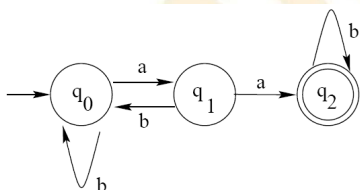
Autómata conexo pero con estados no finalizables

4

Algoritmo para la obtención de los estados finalizables

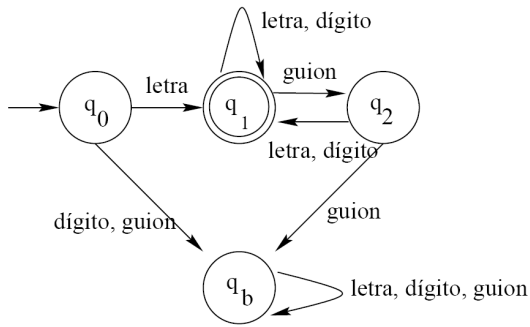
- [1] inicio
- [2] Viejo $\leftarrow \emptyset$
- [3] Nuevo $\leftarrow F$
- [4] mientras (Viejo \neq Nuevo) hacer
- [5] Viejo \leftarrow Nuevo
- [6] para cada $q \notin$ Viejo hacer
- [7] si $\exists \sigma \in \Sigma, \delta(q, \sigma) \in$ Nuevo
- [8] entonces Nuevo \leftarrow Nuevo $\cup \{q\}$
- [9] fin si
- [10] fin para
- [11] fin mientras
- [12] Finalizables \leftarrow Nuevo
- [13] fin

5



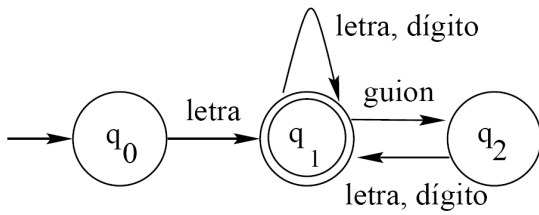
Autómata sin estados inútiles

6



Autómata con un estado inútil

7



Autómata sin estados inútiles

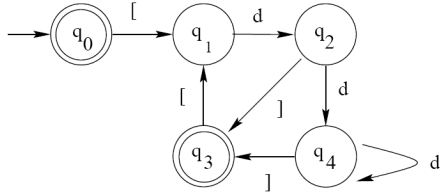
8

Algoritmo para obtener el "Autómata Cociente"

- [1] inicio
- [2] $p_0 \leftarrow Q - F$
- [3] $p_1 \leftarrow F$
- [4] $Nuevo \leftarrow \{p_0, p_1\}$ y p_0 y p_1 , no marcados
- [5] mientras haya un estado $p \in Nuevo$ no marcado hacer
- [6] Marcar a p
- [7] para cada $\sigma \in \Sigma$ hacer
- [8] Dividir p en subconjuntos tales que q_i y q_j estarán
- [9] en el mismo subconjunto si $\delta(q_i, \sigma)$ y $\delta(q_j, \sigma)$
- [10] pertenecen al mismo subconjunto de Nuevo
- [11] fin para
- [12] si se ha dividido p en subconjuntos
- [13] entonces
- [14] Sustituir p en Nuevo por los subconjuntos obtenidos
- [15] Desmarcar a todos los estados de Nuevo
- [16] fin si
- [17] fin mientras
- [18] fin

9

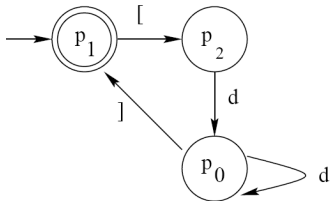
Minimización de un autómata



Autómata no minimizado

10

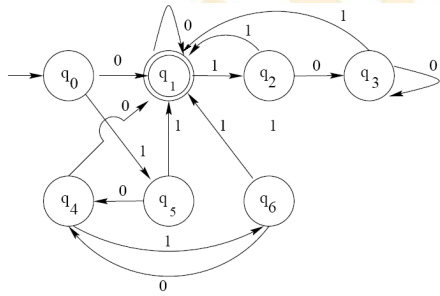
Minimización de un autómata



Autómata minimizado

11

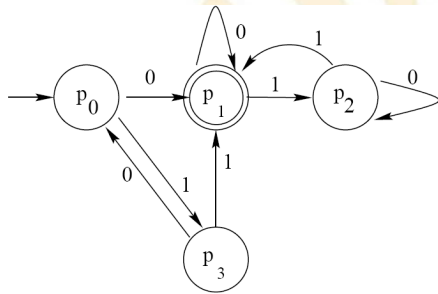
Minimización de un autómata



Autómata no minimizado

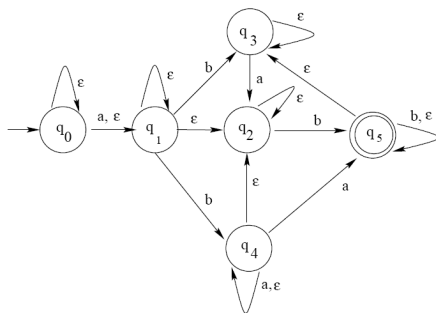
12

Minimización de un autómata



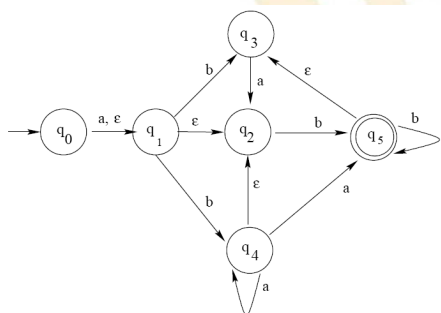
Autómata minimizado

13



Autómata finito no determinista (con transiciones épsilon triviales)

14



Autómata finito no determinista (sin transiciones épsilon triviales)

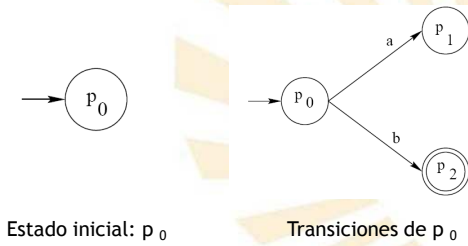
15

Algoritmo de Construcción de Subconjuntos

- [1] inicio
- [2] $p_0 \leftarrow \text{clausura} - \epsilon (q_0)$
- [3] $Q_D \leftarrow \{p_0\}$ y p_0 no marcado
- [4] mientras haya un estado $p \in Q_D$ no marcado hacer
- [5] Marcar a p
- [6] para cada $\sigma \in \Sigma$ hacer
- [7] $p' \leftarrow \text{clausura} - \epsilon (\delta_N(p, \sigma))$
- [8] si $p' \notin Q_D$
- [9] entonces
- [10] $Q_D \leftarrow Q_D \cup \{p'\}$ y p' no marcado
- [11] fin si
- [12] Definir $\delta_D(p, \sigma) \leftarrow p'$
- [13] fin para
- [14] fin mientras
- [15] $F_D \leftarrow \{p_i \mid F_N \cap p_i \neq \emptyset\}$
- [16] fin

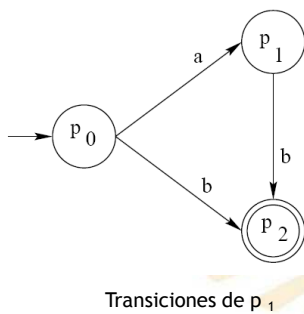
16

Algoritmo de Construcción de Subconjuntos: ejemplo



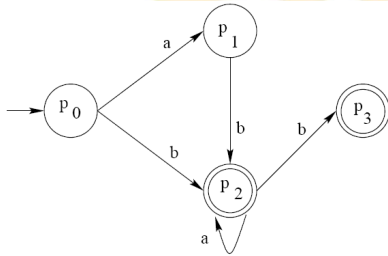
17

Algoritmo de Construcción de Subconjuntos: ejemplo



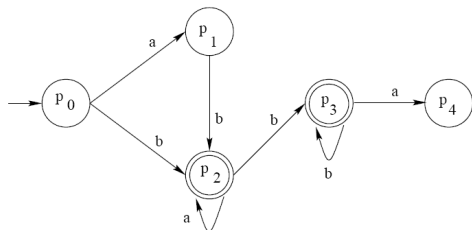
18

Algoritmo de Construcción de Subconjuntos: ejemplo



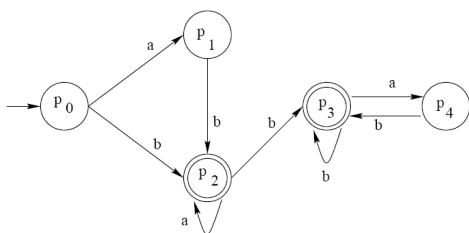
Transiciones de p_2

Algoritmo de Construcción de Subconjuntos: ejemplo



Transiciones de p_3

Algoritmo de Construcción de Subconjuntos: ejemplo



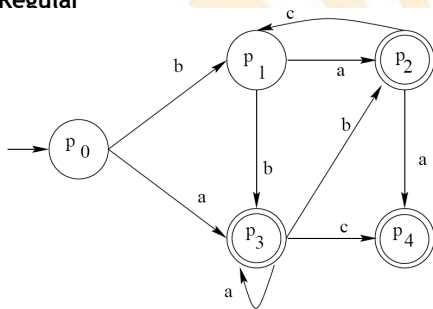
Transiciones de p_4

Ejercicio: transformación de una Gramática regular en un Autómata Finito Determinista

$P = \{$
 $S \rightarrow a A$
 $A \rightarrow a A$
 $A \rightarrow a$
 $A \rightarrow b B$
 $B \rightarrow b B$
 $B \rightarrow b$
 $\}$

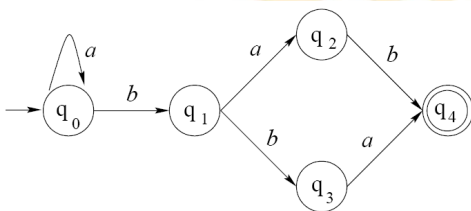
22

Ejercicio: transformación de un Autómata Finito Determinista en una Gramática Regular



23

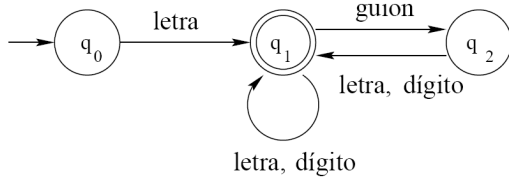
Problema de Análisis
Ejercicio: obtención de la Expresión Regular equivalente a un Autómata Finito Determinista



24

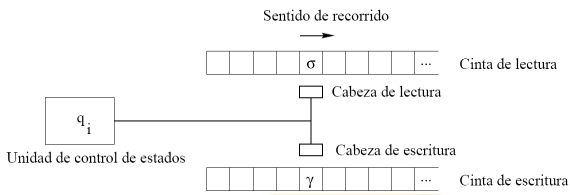
Problema de Análisis

Ejercicio: obtención de la Expresión Regular equivalente a un Autómata Finito Determinista

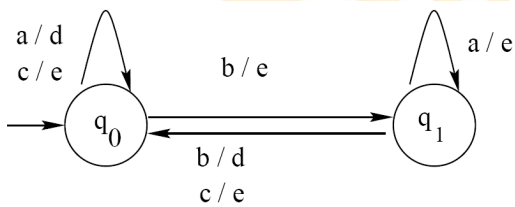


AFD que reconoce identificadores de COBOL

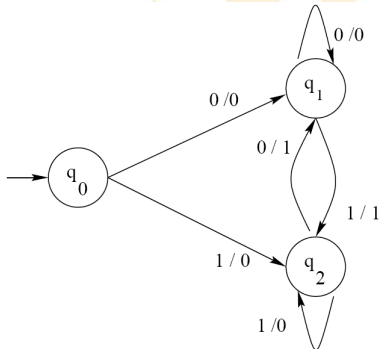
Componentes de una máquina secuencial



Representación gráfica de una máquina secuencial de Mealy

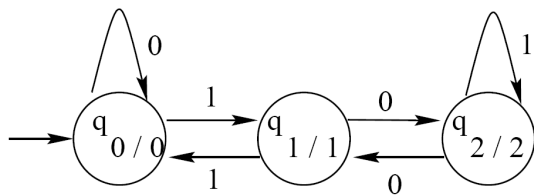


Máquina secuencial de Mealy que detecta



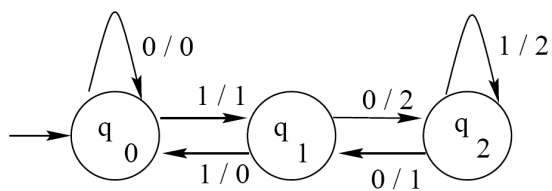
28

Máquina secuencial de Moore que calcula "n mod 3"



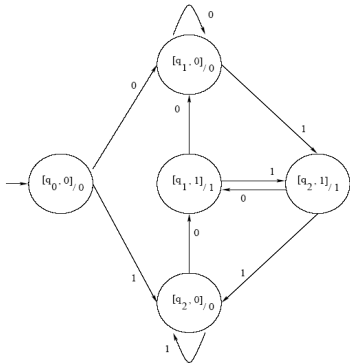
29

Máquina secuencial de Mealy que calcula "n mod 3"



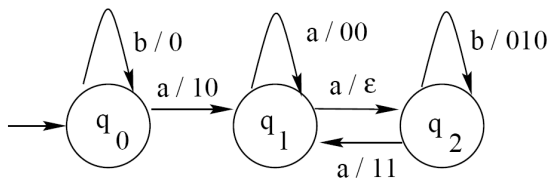
30

Máquina secuencial de Moore que detecta



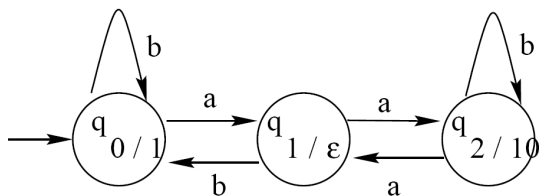
31

Máquina secuencial de Mealy "generalizada"



32

Máquina secuencial de Moore "generalizada"



33



UNIVERSIDAD DE CÓRDOBA
ESCUELA POLITÉCNICA SUPERIOR
DEPARTAMENTO DE INFORMÁTICA Y ANÁLISIS NUMÉRICO
INGENIERÍA TÉCNICA EN INFORMÁTICA DE SISTEMAS
SEGUNDO CURSO, SEGUNDO CUATRIMESTRE



TEORÍA DE AUTÓMATAS Y LENGUAJES FORMALES

**Tema 6.- Autómatas finitos
y máquinas secuenciales**