# Teoría de la Computación y Lenguajes Formales

Prof. Hilda Y. Contreras
Departamento de Computación
hyelitza@ula.ve
hildac.teoriadelacomputacion@gmail.com



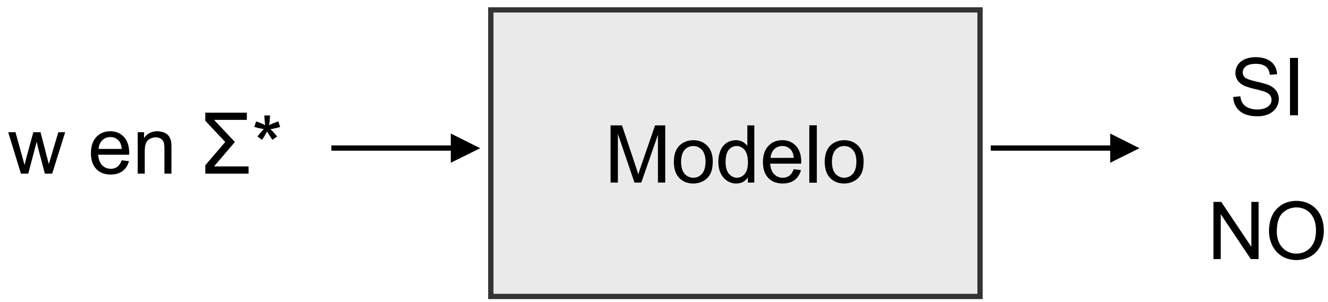
UNIVERSIDAD DE LOS ANDES

# Contenido

**Tema1: Autómatas de estados finitos:**

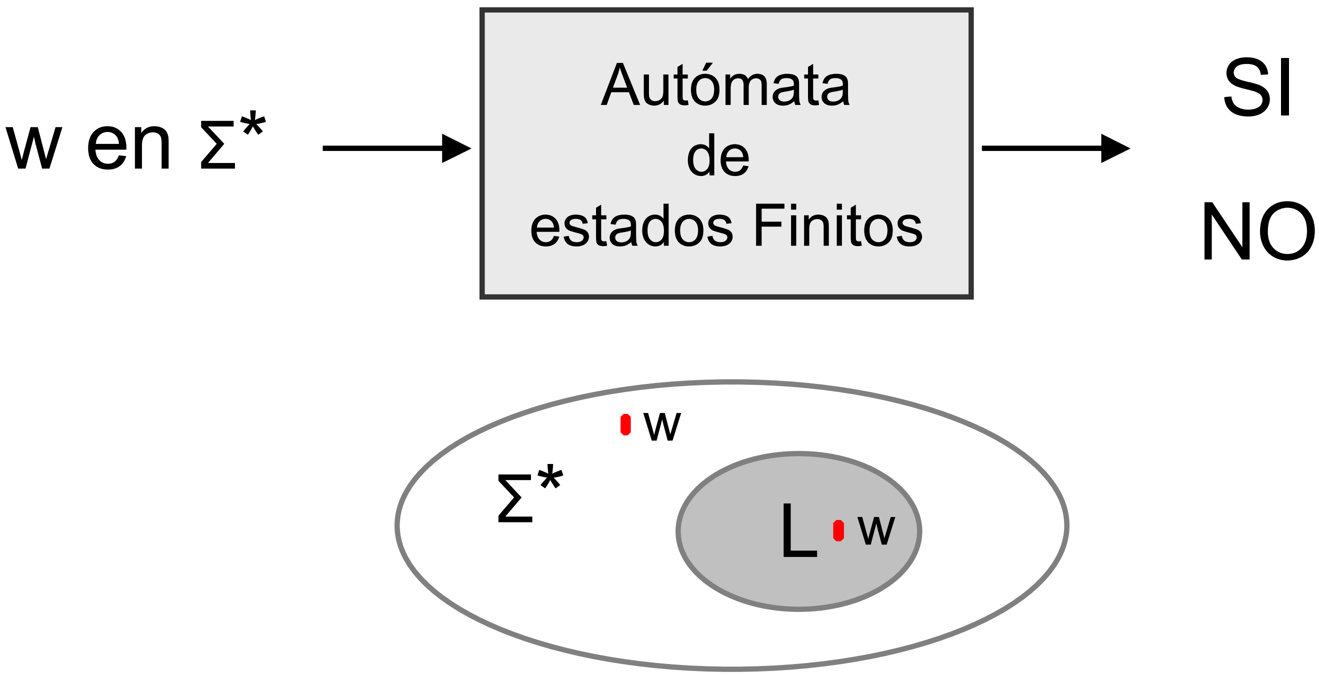
- Modelo para reconocimiento
- Heramienta Jflap
- Autómata Finito Determinista
- Autómata Finito No-Determinista
- Autómata Finito con Transiciones Nulas

# Problema




Definición del Lenguaje L con un Modelo formal

# Modelo de reconocimiento




El Lenguaje L formado por un Alfabeto Σ

# Jerarquía de Lenguajes

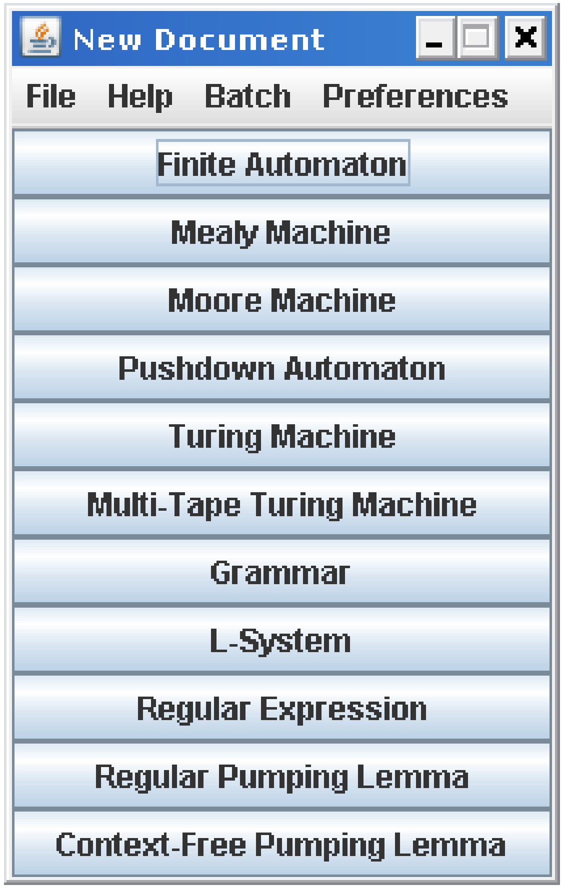
- **Jerarquía de Chomsky (1959):**

| Tipo | Lenguaje | Máquina | Gramática |
|---|---|---|---|
| 0 | Recursivamente enumerable | Máquina de Turing | Sin restricciones |
| 1 | Dependiente del Contexto | Autómata linealmente acotado | $\alpha A\beta \rightarrow \alpha\gamma\beta$ |
| 2 | Independiente del Contexto | Autómata de Pila | $A \rightarrow \gamma$ |
| **3** | **Regular** | **Autómata finito** | $\boldsymbol{A \rightarrow aB}$<br>$\boldsymbol{A \rightarrow a}$ |

# Herramienta Jflap

http://www.jflap.org/

JFLAP




Aplicación educativa para la enseñanza de Teoría de Autómatas y Lenguajes Formales

Aplicación Java

# Autómata de Estados Finitos

Un Autómata Finito es una quíntupla
$M = (Q, \Sigma, \delta, q_0, F)$ donde

$Q$ es un conjunto finito de estados

$\Sigma$ alfabeto de entrada (conjunto finito de símbolos de entrada)

$\delta$ es una aplicación llamada función de transición

$q_0$ es un elemento de $Q$ llamado estado inicial

$F$ es un subconjunto de $Q$, llamado conjunto de estados finales o de aceptación

# Autómata de Estados Finitos

Clasificación (función de transición δ):

- Autómata Finito Determinísta (AFD)
- Autómata Finito No Determinísta (AFND)
- Autómata Finito No Determinísta con transiciones nulas (AFND-λ)

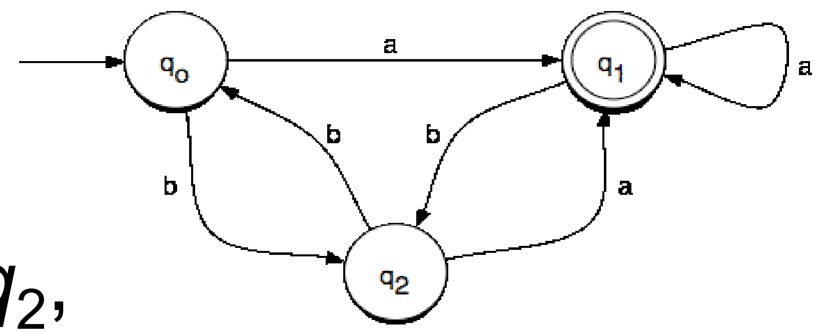
# Autómata Finito Determinista (AFD)

## La función de transición:

$\boldsymbol{\delta : Q \times \Sigma \rightarrow Q}$

Sea el autómata $M = (Q, \Sigma, \delta, q_0, F)$, donde

$Q = \{q_0, q_1, q_2\}$




$\Sigma = \{a, b\}$ y $F = \{q_1\}$

$\delta(q_0, a) = q_1, \quad \delta(q_0, b) = q_2,$

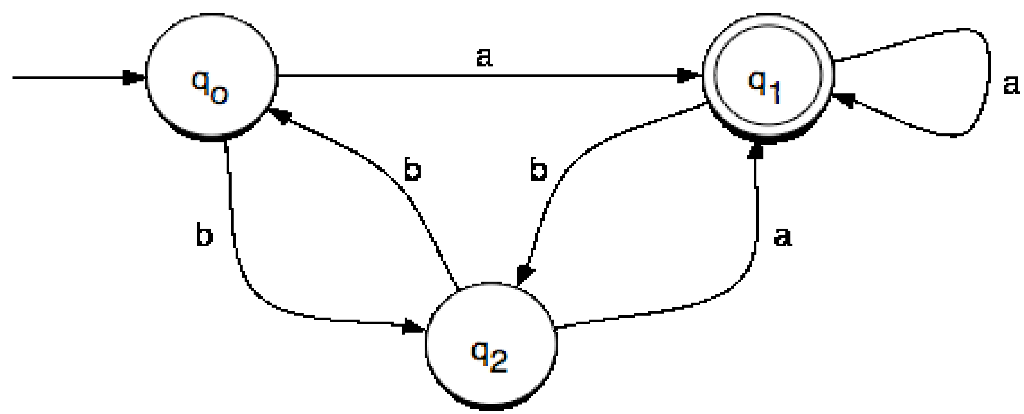
$\delta(q_1, a) = q_1, \quad \delta(q_1, b) = q_2,$

$\delta(q_2, a) = q_1, \quad \delta(q_2, b) = q_0,$

# AFD: Diagrama de transición

Es un grafo en el que:

- Hay un nodo por cada estado
- Por cada transición $\delta(q, a) = p$ hay un arco de $q$ a $p$ con la etiqueta $a$. Donde a pertenece a $\Sigma$
- El estado inicial está indicado con una flecha entrante. Los estados finales están indicados con una doble circunferencia.

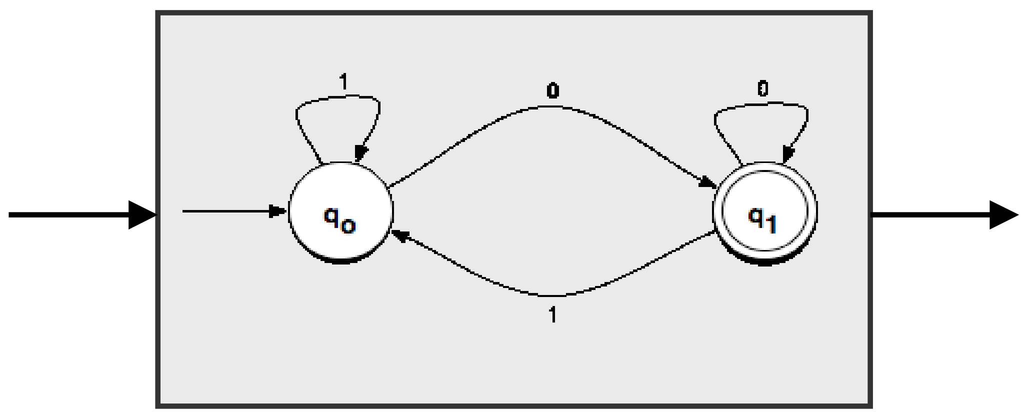


# Funcionamiento del AFD

Entrada: w en Σ*

Salida: si o no (acepta o reconoce la cadena w en L)

w en Σ*



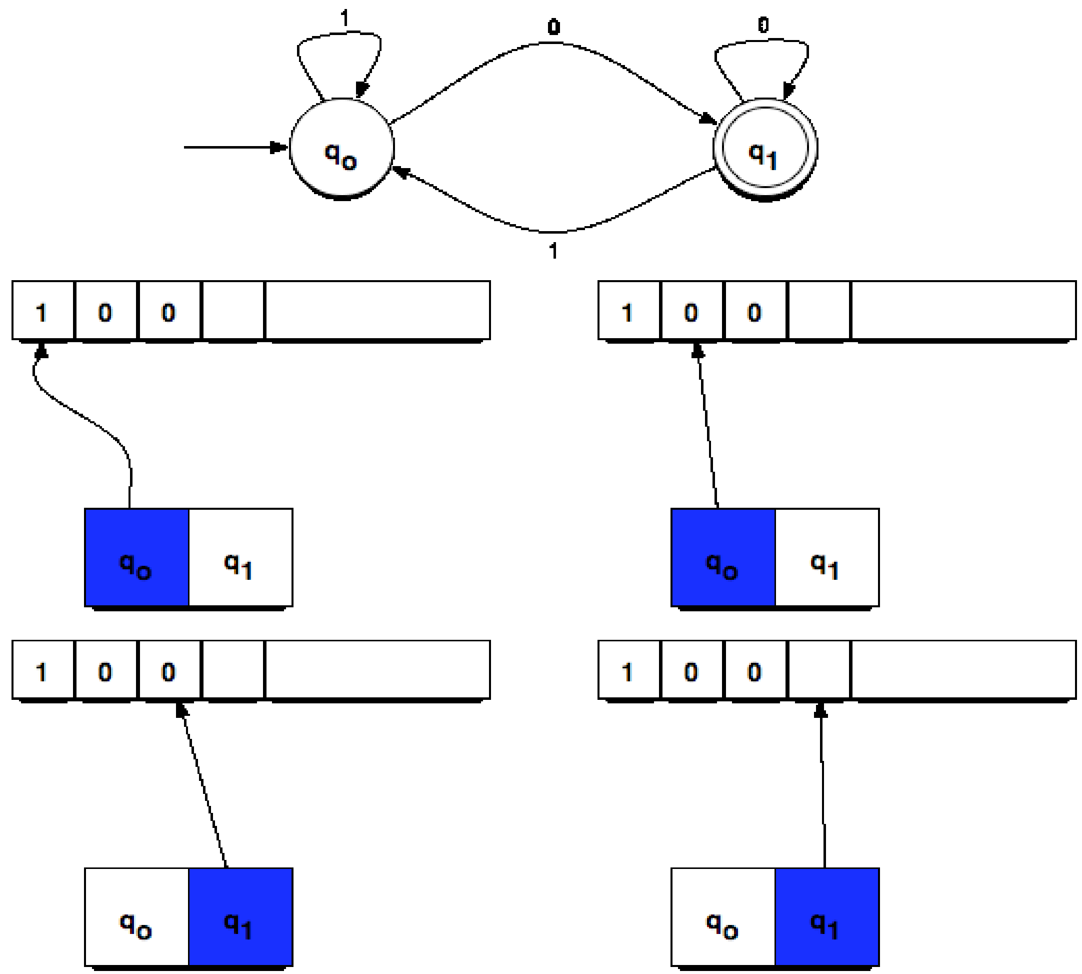

SI

NO

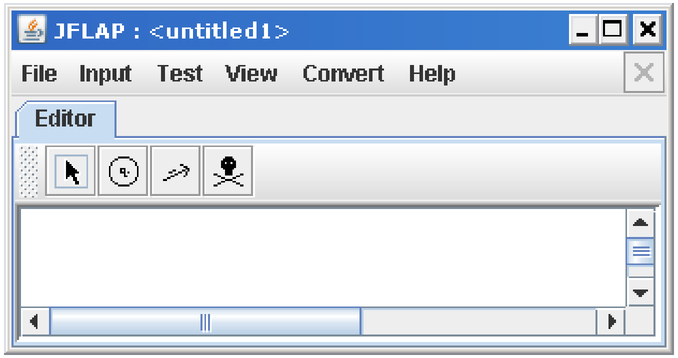
# Funcionamiento del AFD

**JFLAP**

AFD: AFD-1.jff



# Uso de JFLap




La finalidad de los cuatro primeros botones es la siguiente:

- Características de los elementos (botón derecho del ratón sobre el elemento, estado o transición): designar estados como inicial o final, o cambiar el carácter de transición.
- Creación de estados: botón izquierdo del ratón.
- Creación de transiciones: click en el estado inicial, arrastrar hasta el estado final y soltar.
- Borrado de elementos: botón izquierdo del ratón sobre el elemento, estado o transición.

# AFD: Proceso de Cálculo

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

Descripción Instantánea o Configuración:

Un elemento de $Q \times \Sigma^*$ : $(q, \mathrm{w})$.

Configuración Inicial para a en $\Sigma^*$: $(q_0, w)$

Relación paso de cálculo entre dos configuraciones

- $((q, w) \vdash (p, v)) \leftrightarrow ((w = av)$ y $\delta(q, a) = p)$

De una configuración sólo se puede pasar a lo máximo a una configuración en un paso de cálculo.

# AFD: Proceso de Cálculo

Relación de cálculo entre dos configuraciones:

(($q$, w) |-$^{*}$ ($p$, $v$)) si y sólo si existe una sucesión de configuraciones $C_0$, . . . , $C_n$ tales que

$C_0$ = ($q$, w), $C_n$ = ($p$,$v$) y Para todo $i \leq n - 1$, $C_i$ |-$C_{i+1}$

**Lenguaje Aceptado por un AFD**

$$L(M) = \{ w \text{ en } \Sigma^* : (q_0, w) \vdash^* (q, \lambda), q \text{ en } F \}$$

Las palabras de $L(M)$ se dicen aceptadas por el autómata M

# AFD: Proceso de Cálculo

Función de transición extendida $\boldsymbol{\delta^* : Q \times \Sigma^* \rightarrow Q}$

$\delta^*(q,\lambda) = q$, $\lambda$ la cadena vacía no cambia de estado

$\delta^*(q,w) = \delta(\delta^*(q,v),a)$, donde $w,v$ en $\Sigma^*$, $a$ en $\Sigma$ y $w=va$

**Lenguaje Aceptado por un AFD**

$$\boxed{L(M) = \{\ w \text{ en } \Sigma^* : \delta^*(q_0, w) \text{ en } F\ \}}$$

Las palabras de *L*(*M*) se dicen aceptadas por el autómata M

# AFD: Ejemplo de Cálculo

$M = (\{q_0,q_1\}, \{0,1\}, \delta, q_0, \{q_1\})$

Para la cadena w = **100** en $\Sigma^*$

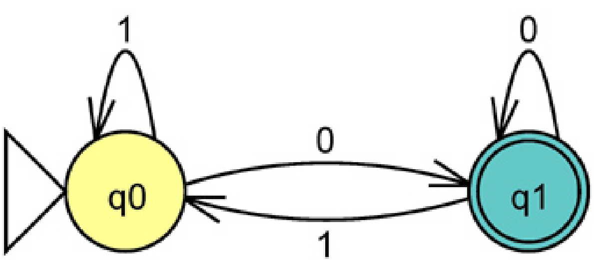
- Con las descripciones instantáneas

$$\mathbf{(q_0,\underline{1}00)} \vdash (q_0,\underline{0}0) \vdash (q_1,\underline{0}) \vdash \mathbf{(q_1,\lambda)}$$

Si acepta la cadena en L ($q_1$ en F)

- Con la función extendida:

$$\begin{aligned}\boldsymbol{\delta^*(q_0,100)} &= \delta(\delta(\delta(\delta^*(q_0, \boldsymbol{\lambda}),1),0),0) \\ &= \delta(\delta(\delta(q_0,1),0),0) \\ &= \delta(\delta(q_0,0),0) \\ &= \delta(q_1,0) \\ &= \mathbf{q}_1 \text{ en F (acepta)}\end{aligned}$$



# AFD: Ejemplo de Cálculo

$M = (\{q_0,q_1\}, \{0,1\}, \delta, q_0, \{q_1\})$

- Para la cadena w = **1101** en $\Sigma^*$

  $\mathbf{(q_0,\underline{1}101)} \vdash (q_0,\underline{1}01) \vdash (q_0,\underline{0}1) \vdash (q_1,\underline{1}) \vdash \mathbf{(q_0, \lambda)}$

  No acepta la cadena en L ($q_0$ no está en F)

- Con la función extendida:

  $\boldsymbol{\delta^*(q_0,1101)} = \delta(\delta(\delta(\delta(\delta^*(q_0, \lambda),1),1),0),1)$

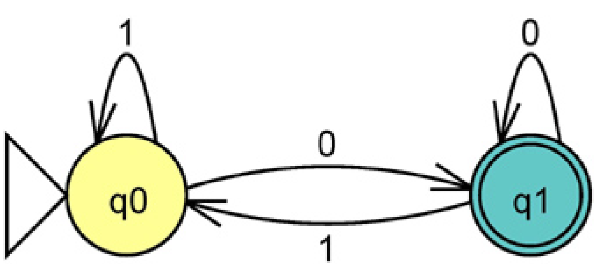
  $= \delta(\delta(\delta(\delta(q_0,1),1),0),1)$

  $= \delta(\delta(\delta(q_0,1),0),1)$

  $= \delta(\delta(q_0,0),1)$

  $= \delta(q_1,1)$

  $= \mathbf{q}_0$ no esta en F (No acepta)



# AFD: Tabla de transición

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

$Q = \{q_0, q_1\}$

$\Sigma = \{0, 1\}$

$F = \{q_1\}$

$\delta(q_0, 0) = q_1,$

$\delta(q_0, 1) = q_0,$

$\delta(q_1, 0) = q_1,$

$\delta(q_1, 1) = q_0,$

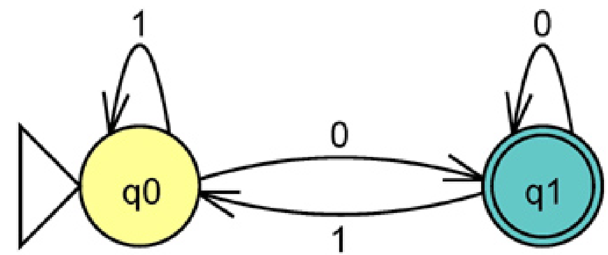



Tabla de transición

| | | $\Sigma$ | |
|---|---|---|---|
| | $\delta$ | **0** | **1** |
| Q | $\mathbf{q_0}$ | $q_1$ | $q_0$ |
| | $\mathbf{q_1}$ | $q_1$ | $q_0$ |

# Autómata Finito NO Determinista (AFND)

**La función de transición:**

**$\delta : Q \times \Sigma \rightarrow Q^*$**

Sea el autómata $M = (Q, \Sigma, \delta, q_0, F)$, donde

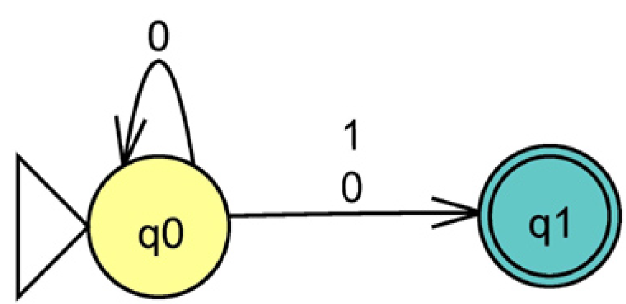
$Q = \{q_0, q_1\}$

$\Sigma = \{0, 1\}$ y $F = \{q_1\}$

Rango de $\delta$ es $Q^* = \{\phi, \{q_0\}, \{q_1\}, \{q_0, q_1\}\}$

$\delta(q_0, 0) = \{q_1, q_0\}$

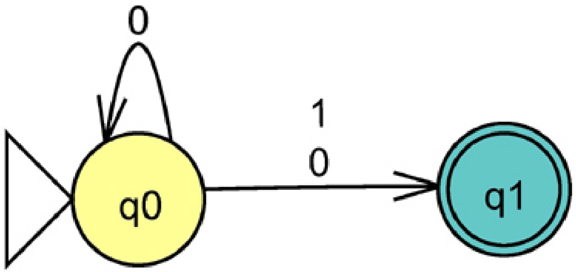
$\delta(q_0, 1) = \{q_1\}$

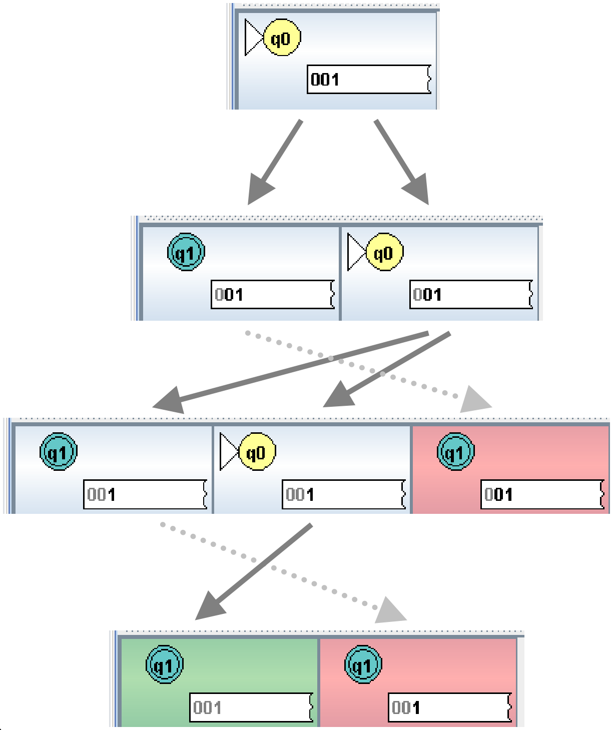


# Funcionamiento del AFND

**JFLAP**

AFND: AFND-1.jff



# AFND: Tabla de transición

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

$Q = \{q_0, q_1\}$

$\Sigma = \{0, 1\}$

$\mathrm{F} = \{q_1\}$

$\delta(q_0, 0) = \{q_0, q_1\}$

$\delta(q_0, 1) = \{q_1\}$

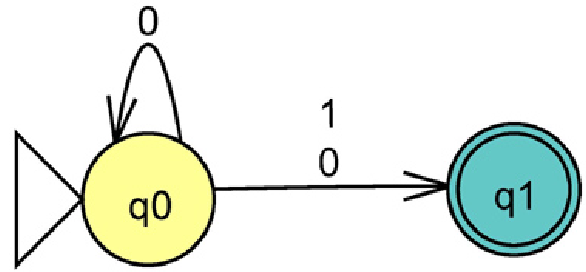



Tabla de transición

Σ

Q

| δ | **0** | **1** |
|---|---|---|
| $\mathbf{q_0}$ | $\{q_0,q_1\}$ | $\{q_0\}$ |
| $\mathbf{q_1}$ | $\Phi$ | $\Phi$ |

# AFND: Proceso de Cálculo

Función de transición extendida **$\delta^*$: *$Q \times \Sigma^* \rightarrow Q^*$***

$\delta^*(q,\lambda) = \{q\}$, $\lambda$ la cadena vacía no cambia de estado

$\delta^*(q,w) = \delta(\delta^*(q,a),v)$, donde w,v en $\Sigma^*$, a en $\Sigma$ y w=av

- $\delta^*(q,w) = \delta^*(\{q\},w)$
- Si *B* es subconjunto de $Q \rightarrow \delta^*(B,a) = \text{U } \delta(q,a)$ para todo q *en B*

**Lenguaje Aceptado por un AFND**

$$\boxed{L(M) = \{ w \text{ en } \Sigma^* : \delta^*(q_0, w) \cap \text{ con } F \neq \Phi\}}$$

Las palabras de *L*(*M*) se dicen aceptadas por el autómata M

# AFND: Ejemplo de Cálculo

$M = (\{q_0,q_1\}, \{0,1\}, \delta, q_0, \{q_1\})$

Para la cadena w = **001** en $\Sigma^*$

- Con las descripciones instantáneas

$\mathbf{(q_0,\underline{0}01)}$ |- $(q_1,\underline{0}1)$ X

|- $(q_0,\underline{0}1)$ |- $(q_1,\underline{1})$ X

|- $(q_0,\underline{1})$ |- $\mathbf{(q_1,\lambda)}$ Si acepta 001 en L ($q_1$ en F)

- Con la función extendida:

$\boldsymbol{\delta^*(q_0,001)} = \delta(\delta(\delta(\delta^*(\{q_0\}, \lambda),0),0),1)$

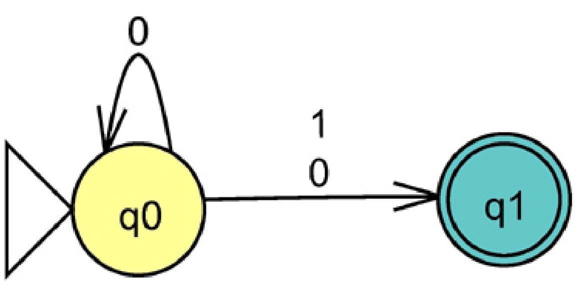
$= \delta(\delta(\delta(\{q_0\},0),0),1)$

$= \delta(\delta(\{q_1,q_0\},0),1)$

$= \delta(\{q_1,q_0\},1)$

$= \{\mathbf{q_1}\} \cap$ con $F \neq \Phi$

(acepta 001 en L)



# AFND → AFD

- Dado un autómata no determinista se le hace corresponder uno determinista que recorre todos los caminos al mismo tiempo
- Un autómata no-determinista y su determinista asociado aceptan el mismo lenguaje (son equivalentes)

**Teorema**: Si L es aceptado por un AFN N entonces existe un AFD D que acepta L (es decir, ambos aceptan la misma clase de lenguaje, L(N) = L(D)).

# AFND → AFD

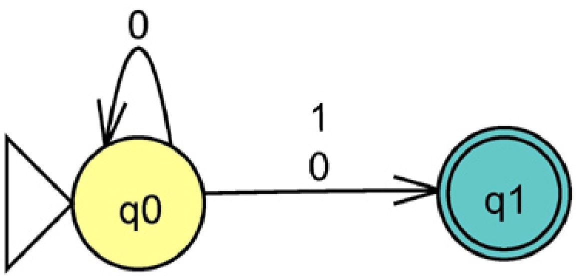
Dado el AFND $N = (Q_N, \Sigma, \delta_N, q_0, F_N)$ se quiere obtener un AFD $D = (Q_D, \Sigma, \delta_D, \{q_0\}, F_D)$ equivalente:

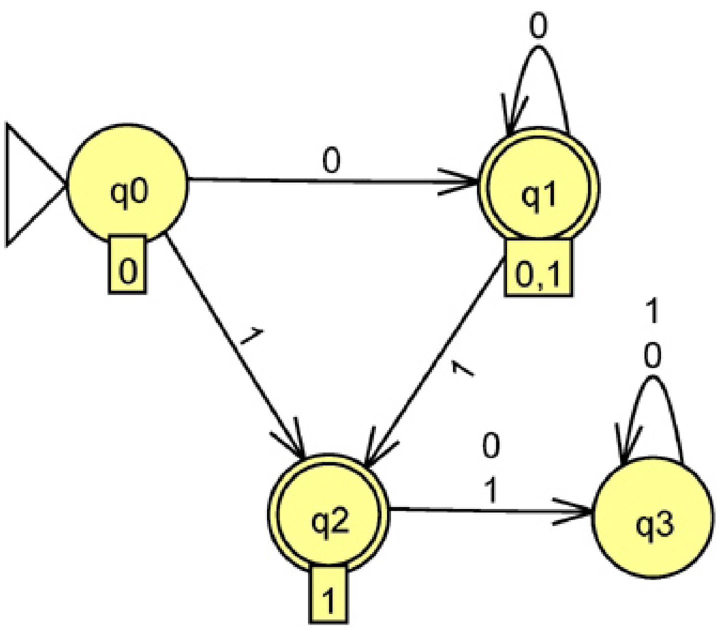
Algoritmo:

- $\Sigma$ de N y D son iguales.
- $Q_D = Q_N^*$ (conjunto potencia). Sólo considera los estados de $Q_D$ accesibles desde $q_0$.
- $F_D$ es el conjunto de subconjuntos S de $Q_N$ (S subconjunto de $Q_N$) tal que $S \cap F_N \neq \phi$ . Es decir $F_D$ contiene todos los conjuntos de estados de N que incluyen al menos un estado de aceptación de N.
- Para todo S subconjunto de $Q_N$ y a en $\Sigma$ , entonces $\delta_D(S,a) = \bigcup \delta_N(p,a)$, para todo p en S.

# AFND → AFD

- Ejemplo:




| | $\delta_D$ | 0 | 1 |
|---|---|---|---|
| $q_0$ | $\{q_0\}$ | $\{q_0,q_1\}$ | $\{q_1\}$ |
| $q_1$ | $\{q_0,q_1\}$ | $\{q_0,q_1\}$ | $\{q_1\}$ |
| $q_2$ | $\{q_1\}$ | $\{\}$ | $\{\}$ |
| $q_3$ | $\{\}$ | $\{\}$ | $\{\}$ |

# Autómata Finito No Determinista con transiciones nulas (AFND- $\lambda$)

**La función de transición:**

$\boldsymbol{\delta : Q \times \Sigma \cup \{\lambda\} \rightarrow Q^*}$

Sea el autómata $M = (Q, \Sigma, \delta, q_0, F)$, donde

$Q = \{q_0, q_1, q_2\}$

$\Sigma = \{0, 1\}$ y $F = \{q_2\}$

Rango de δ es Q*

$Q^* = \{\phi,\{q_0\},\{q_1\}, \{q_2\},\{q_0,q_1\},\{q_0,q_2\},\{q_2,q_1\}, \{q_0,q_1,q_2\}\}$

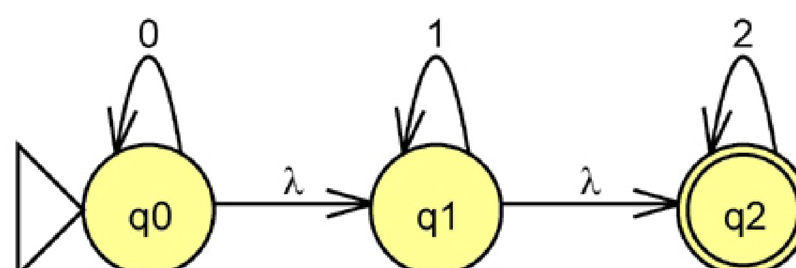
$\delta(q_0, 0) = \{q_0\}$

$\delta(q_1, 1) = \{q_1\}$

$\delta(q_2, 1) = \{q_2\}$

$\delta(q_0, \lambda) = \{q_1\}$

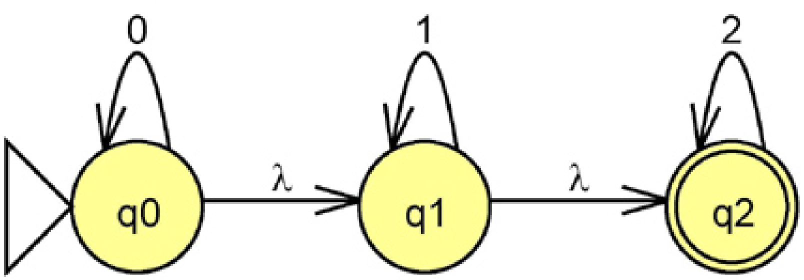
$\delta(q_1, \lambda) = \{q_2\}$

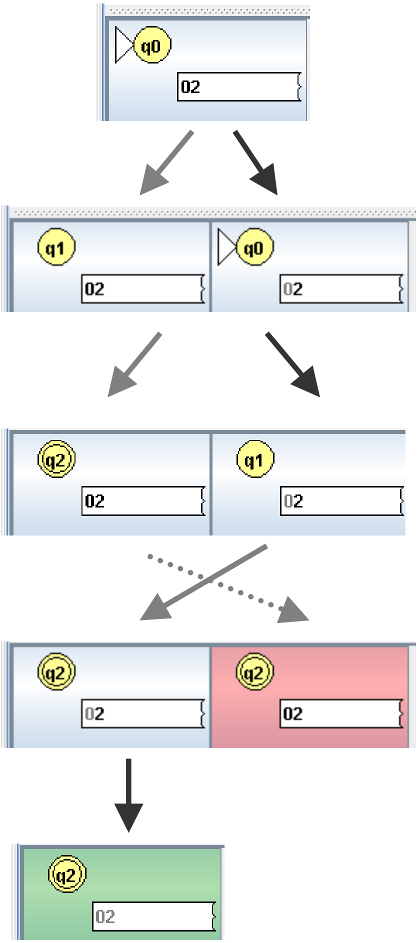


# Funcionamiento del AFND-λ

**JFLAP**

AFND-λ: AFNDE-1.jff



# AFND-λ : Tabla de transición

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

$Q = \{q_0, q_1, q_2\}$

$\Sigma = \{0, 1\}$

$\mathrm{F} = \{q_2\}$

$\delta(q_0, 0) = \{q_0\}$

$\delta(q_1, 1) = \{q_1\}$

$\delta(q_2, 1) = \{q_2\}$

$\delta(q_0, \lambda) = \{q_1\}$

$\delta(q_1, \lambda) = \{q_2\}$

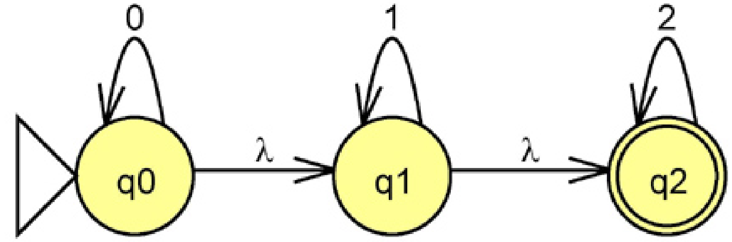



Tabla de transición

$\Sigma$

| | δ | **0** | **1** | **2** | **λ** |
|---|---|---|---|---|---|
| | $\mathbf{q_0}$ | $\{q_0\}$ | $\{\}$ | $\{\}$ | $\{q_1\}$ |
| Q | $\mathbf{q_1}$ | $\{\}$ | $\{q_1\}$ | $\{\}$ | $\{q_2\}$ |
| | $\mathbf{q_2}$ | $\{\}$ | $\{\}$ | $\{q_2\}$ | $\{\}$ |

# AFND-λ: Proceso de Cálculo

De una configuración se puede pasar a varias configuraciones distintas en un paso de cálculo, e incluso a ninguna

Función de transición extendida $\delta^*$: $\boldsymbol{Q \times \Sigma^* \rightarrow Q^*}$

Definición recursiva de λ-clausura(q): Q → Q*

<u>Base</u>: λ-clausura(q) = q , es decir por definición $\delta(q, \lambda) = q$, podemos imaginarnos un bucle de un estado a si mismo etiquetado λ

<u>Paso inductivo</u>:

Si δ es la función de transición de AFN-λ y p está en la λ-clausura(q), es decir $\delta(q, \lambda)=p$ entonces la λ-clausura(q) también contiene los estados $\delta(p,\lambda)$.

# AFND-λ: Proceso de Cálculo

Función de transición extendida **$\delta^*$: $Q \times \Sigma^* \rightarrow Q^*$**

$\delta^*(q,\lambda)$ = λ-clausura(q)

$\delta^*(q,w)$ = λ-clausura($\delta(\delta^*(q,a),v)$), donde w,v en $\Sigma^*$, a en $\Sigma$ y w=av

- $\delta^*(q,w) = \delta^*(\{q\},w)$

Si *B* es subconjunto de *Q entonces:*

- $\delta^*(B,a) = \bigcup \delta(q,a)$ para todo q *en B*
- λ-clausura(B) = $\bigcup$ λ-clausura(q) para todo q *en B*

# AFND-λ: Proceso de Cálculo

**Lenguaje Aceptado por un AFND-λ**

$$L(M) = \{ \text{ w en } \Sigma^* : \delta^*(q_0, w) \cap \text{ con } F \neq \Phi \}$$

Las palabras o cadenas de *L*(*M*) se dicen aceptadas por el autómata M

# AFND-λ: Ejemplo de Cálculo

$$M = (\{q_0,q_1,q_2\}, \{0,1,2\}, \delta, q_0, \{q_2\})$$

Para la cadena w = **02** en $\Sigma^*$

- Con las descripciones instantáneas

$(\mathbf{q_0,\underline{0}2})$ $\begin{cases} \vdash (q_1,\underline{0}2) \vdash (q_2,\underline{0}2) \text{ X} \\ \vdash (q_0,\underline{2}) \vdash (q_1,\underline{2}) \vdash (q_2,\underline{2}) \vdash \mathbf{(q_2, \lambda)} \end{cases}$ Si acepta 02 en L ($q_2$ en F)

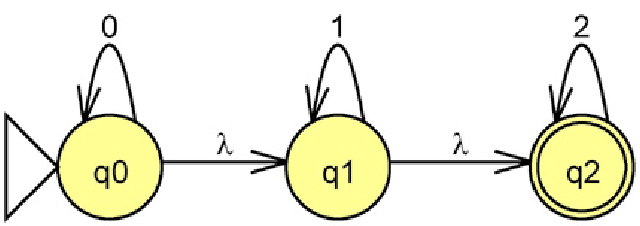
- Con la función extendida:

$$\boldsymbol{\delta^*(q_0,02)} = \delta(\delta(\delta^*(\{q_0\}, \boldsymbol{\lambda}),0),2)$$
$$= \delta(\delta(\{q_0,q_1,q_2\},0),2)$$
$$= \delta(\{q_0,q_1,q_2\},2)$$
$$= \{\mathbf{q}_2\} \quad \cap \text{ con F} \neq \Phi \text{ (acepta 02 en L)}$$

λ -clausura($q_0$) = $\{q_0,q_1,q_2\}$

λ -clausura($q_1$) = $\{q_1,q_2\}$

λ -clausura($q_2$) = $\{q_2\}$



# AFND-$\lambda$ → AFD

- Dado un autómata no determinista con transiciones nulas se le hace corresponder uno determinista que recorre todos los caminos al mismo tiempo
- Un autómata no-determinista con transiciones nulas y su determinista asociado aceptan el mismo lenguaje (son equivalentes)

**Teorema**: Si L es aceptado por un AFND-$\lambda$ N entonces existe un AFD D que acepta L (es decir, ambos aceptan la misma clase de lenguaje, L(N) = L(D)).

# AFND-λ → AFD

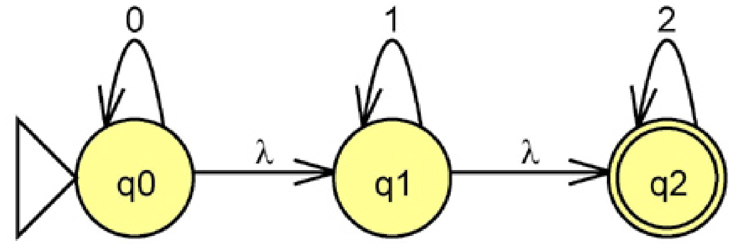
Dado el AFND-λ $N = (Q_N, \Sigma, \delta_N, q_0, F_N)$ se quiere obtener un AFD $D = (Q_D, \Sigma, \delta_D, q_{0D}, F_D)$ equivalente:

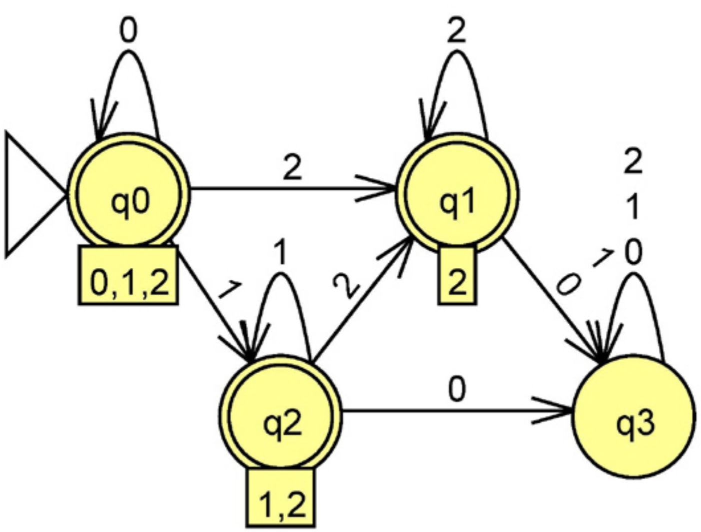
Algoritmo:

- $q_{0D} = \lambda\text{-clausura}(q_0)$
- $\Sigma$ de N y D son iguales.
- $Q_D = Q_N^*$ (conjunto potencia). Sólo considera los estados de $Q_D$ accesibles desde $q_0$.
- $F_D$ es el conjunto de subconjuntos S de $Q_N$ (S subconjunto de $Q_N$) tal que $S \cap F_N \neq \phi$ . Es decir $F_D$ contiene todos los conjuntos de estados de N que incluyen al menos un estado de aceptación de N.
- Para todo S subconjunto de $Q_N$ y a en $\Sigma$ , entonces $\delta_D(S,a) = \bigcup \lambda\text{-clausura}(\delta_N(p,a))$, para todo p en S.

# AFND-λ → AFD

- Ejemplo:




| | $\delta_D$ | 0 | 1 | 2 |
|---|---|---|---|---|
| $q_0$ | $\{q_0,q_1,q_2\}$ | $\{q_0,q_1,q_2\}$ | $\{q_1,q_2\}$ | $\{q_2\}$ |
| $q_1$ | $\{q_1,q_2\}$ | $\{\}$ | $\{q_1,q_2\}$ | $\{q_2\}$ |
| $q_2$ | $\{q_2\}$ | $\{\}$ | $\{\}$ | $\{q_2\}$ |
| $q_3$ | $\{\}$ | $\{\}$ | $\{\}$ | $\{\}$ |

# Ejercicios

Modelar un AF en JFLAP que reconozcan los siguientes lenguajes en $\Sigma = \{0,1\}$:

- cadenas que terminan en 00
- cadenas con dos unos consecutivos
- cadenas que no contengan dos unos consecutivos
- cadenas con dos ceros consecutivos o dos unos consecutivos
- cadenas acabadas en 00 o 11
- cadenas con un 1 en la antepenúltima posición
- cadenas de longitud 4