Lenguajes Formales y Autómatas

* Autómatas finitos
* Autómatas finitos deterministas
* Autómatas finitos no deterministas
* Equivalencia entre AFD y AFN

Lenguajes regulares

Tipo	Lenguajes	Tipo de máquina	Normas para la gramática
0	Recursivamente enumerables	Máquina de Turing	No restringida
1	Sensibles al contexto	Autómata lineal acotado	$\alpha \to \beta$, $\|\alpha\| \leq \|\beta\|$
2	Independientes del contexto	Autómata de pila	$A \to \gamma$
3	Regulares	**Autómata finito**	$A \to aB$ $A \to a$

Universidad del Valle

Lenguajes regulares

- $\{a\}^*$
- $\{a\}^* \cup \{b\}^*$
- $\{a\}^* \cdot \{b\}^*$
- $\{a,bc\}^*$
- $\{a\} \cdot \{b,c,ab\}$
- $\{(ab)^n \mid n \geq 0\}$
- $\{a^n b^m \mid n \geq 0,\ m \geq 0\}$
- $\{a^l b^m c^n \mid l \geq 0, m \geq 0, n \geq 0\}$

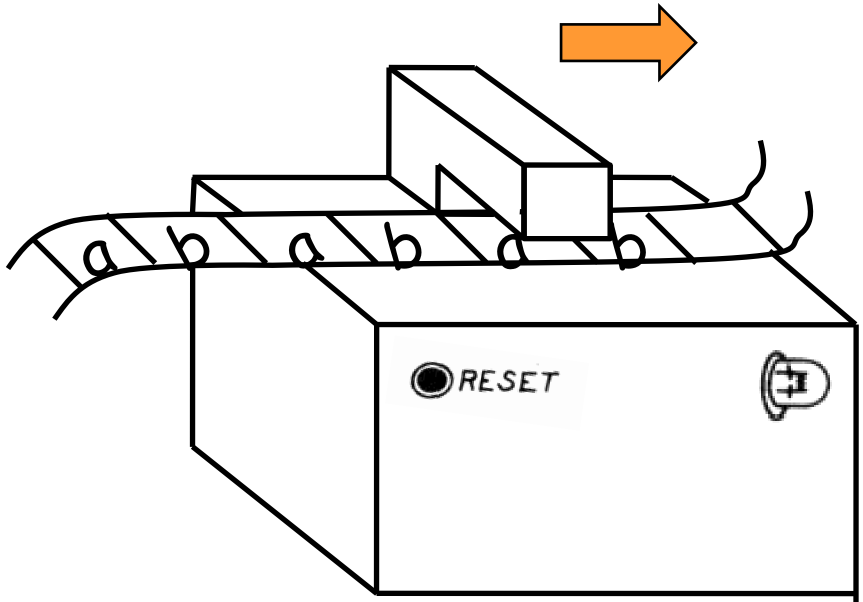
Se puede construir un **autómata finito** para cada uno de estos lenguajes

Lenguajes regulares

- $\{a^nb^n | n \geq 0\}$, no es regular
- $\{a^nb^{2n} | n \geq 0\}$, no es regular
- $\{wcw | w \in \{a,b\}^*\}$, no es regular

No se puede construir un **autómata finito** para ninguno de estos lenguajes

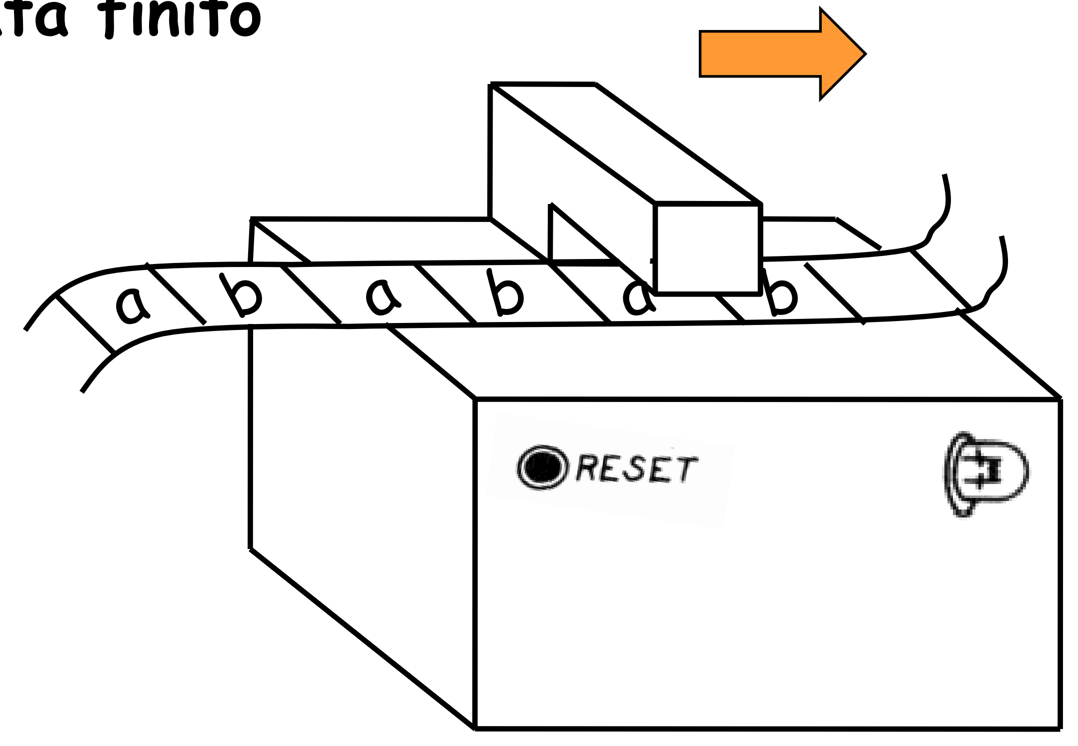
Autómata finito



✓un **autómata** es una **máquina teórica**, es decir, un dispositivo cuyo funcionamiento se estudia sin necesidad de construirlo realmente.

✓Un autómata **lee unas instrucciones** en forma de símbolos y **cambia de estado** según éstas.

Autómata finito



Se puede diseñar un autómata finito para que **acepte**, por ejemplo, el lenguaje $\{ab\}^*=\{\varepsilon,ab,abab,ababab,\ldots\}$

Lenguajes regulares

Considere el lenguaje regular L representado por

$$\mathbf{c^*(a \cup bc^*)^*}$$

- ¿w_1=$\mathbf{abc^5ab}$ pertenece a L?
- ¿w_2=$\mathbf{cabac^3bc}$ pertenece a L?

Lenguajes regulares

Considere el lenguaje regular L representado por

$$\mathbf{c^*(a \cup bc^*)^*}$$

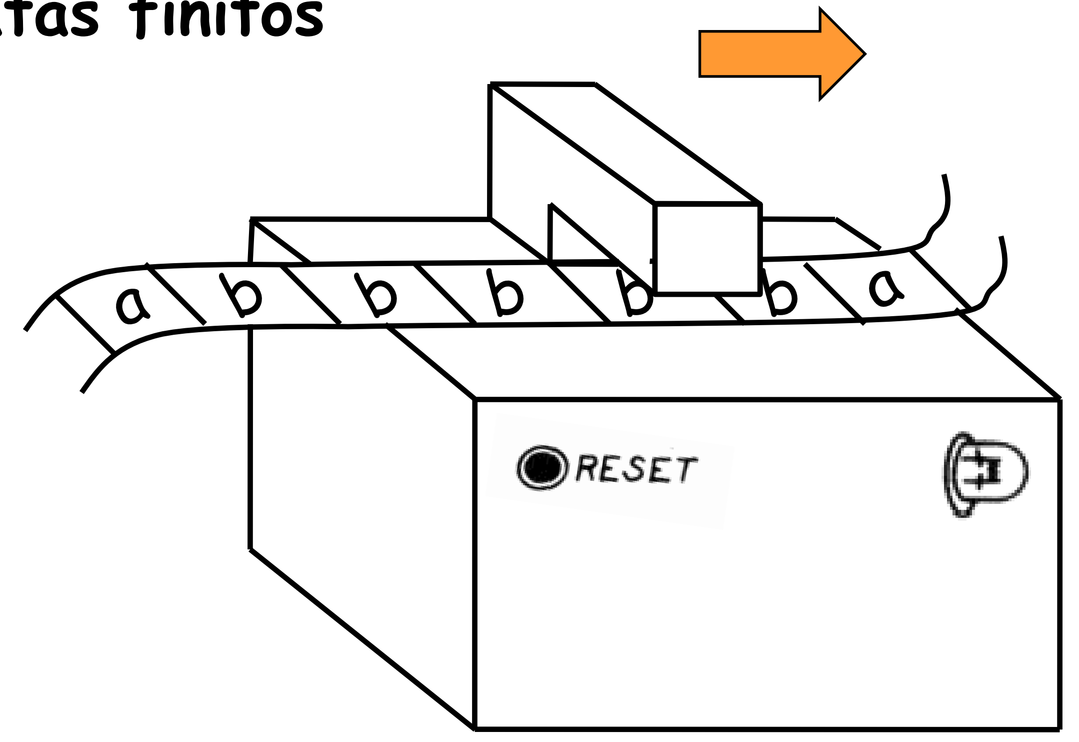
- ¿w_1=$\mathbf{abc^5ab}$ pertenece a L?
- ¿w_2=$\mathbf{cabac^3bc}$ pertenece a L?

Se quiere conocer si una cadena w es generada por un lenguaje L, para esto se puede crear un **autómata finito**

Autómatas finitos

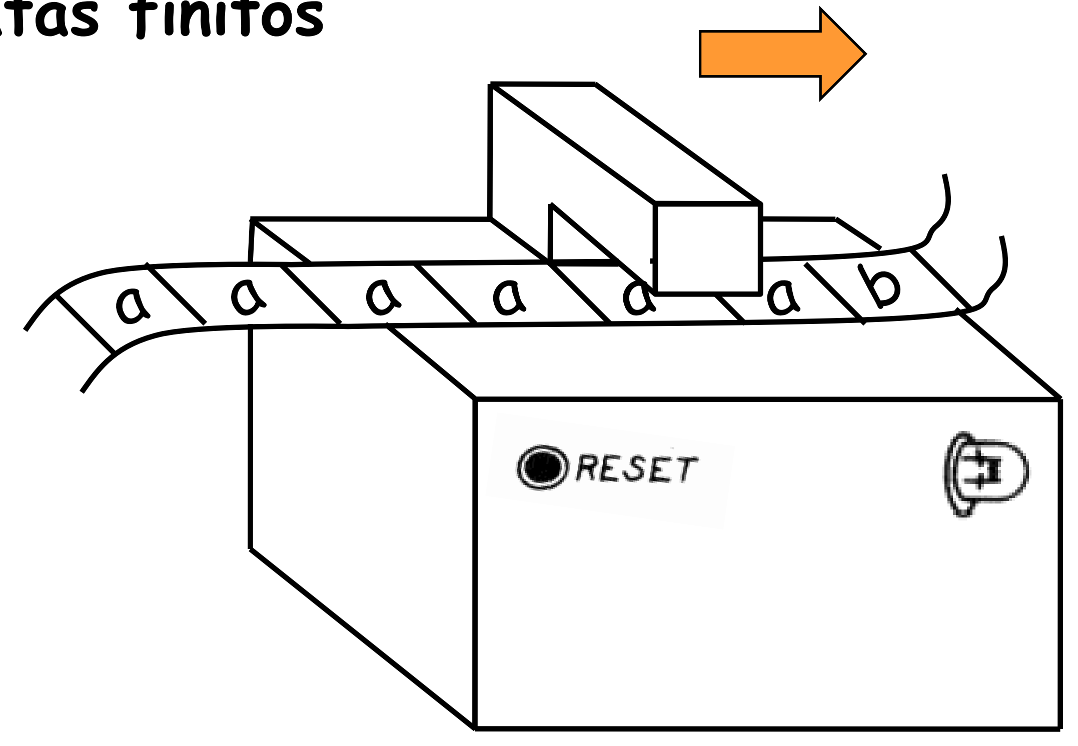
- ✓ Caja negra que acepta como **entrada** los **datos de** una **cinta**
- ✓ Se tiene un bombillo que representa la **salida**, **cuando la entrada se acepta** por el autómata, éste se enciende
- ✓ Botón **reset** para reiniciar
- ✓ La **operación** de la máquina consiste en un conjunto de **estados** internos

Autómatas finitos



La cabeza del autómata sólo puede **leer** (no puede escribir) y se mueve siempre a la **derecha**

Autómatas finitos



Considere un autómata que acepta cadenas en {a,b}*
que tienen una sola **b** y está al final de la cadena

Autómatas finitos

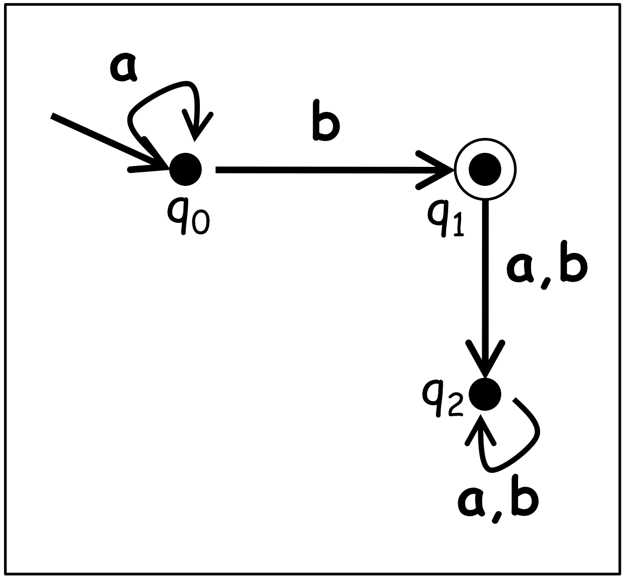
• Los autómatas se pueden representar por medio de un grafo dirigido conocido como **diagrama de transición**

• **Nodos** (**estados**)

Estado inicial

Estado de aceptación

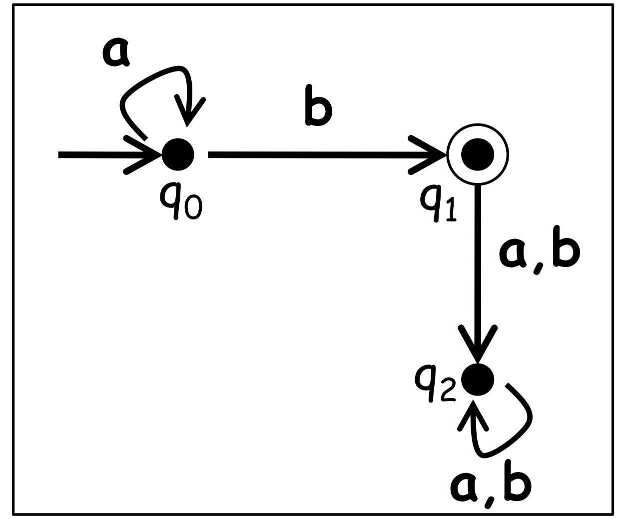
• **Aristas** (**transiciones**)

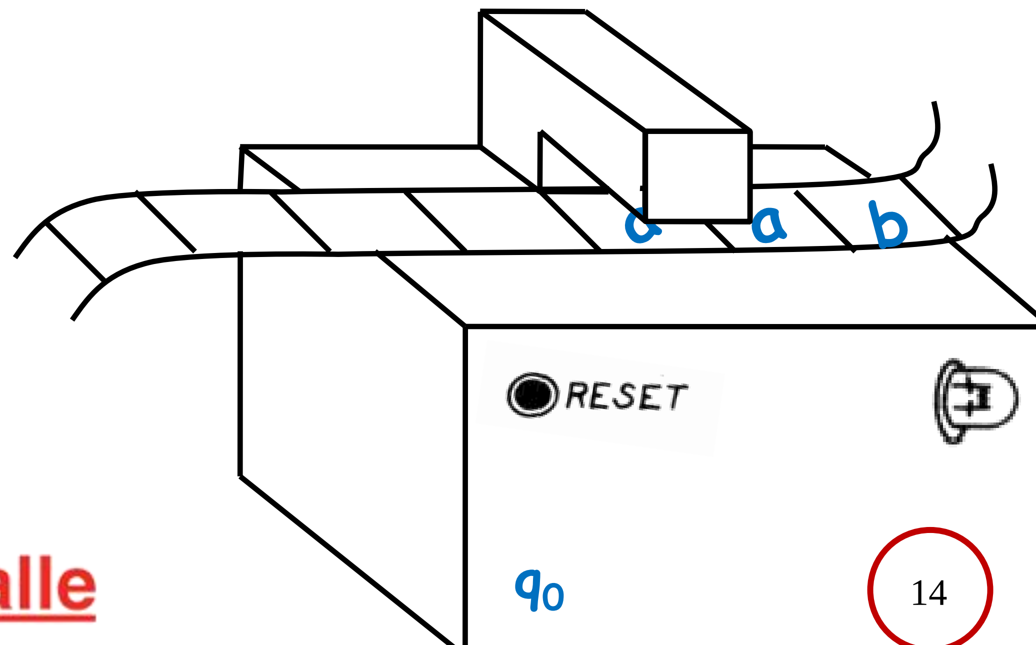


Lenguajes regulares

Cada **avance** en el autómata depende de:

(simboloLeido, estadoActual)

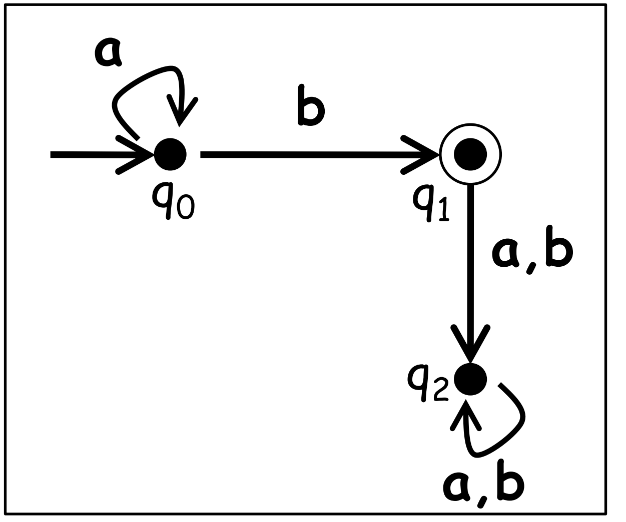


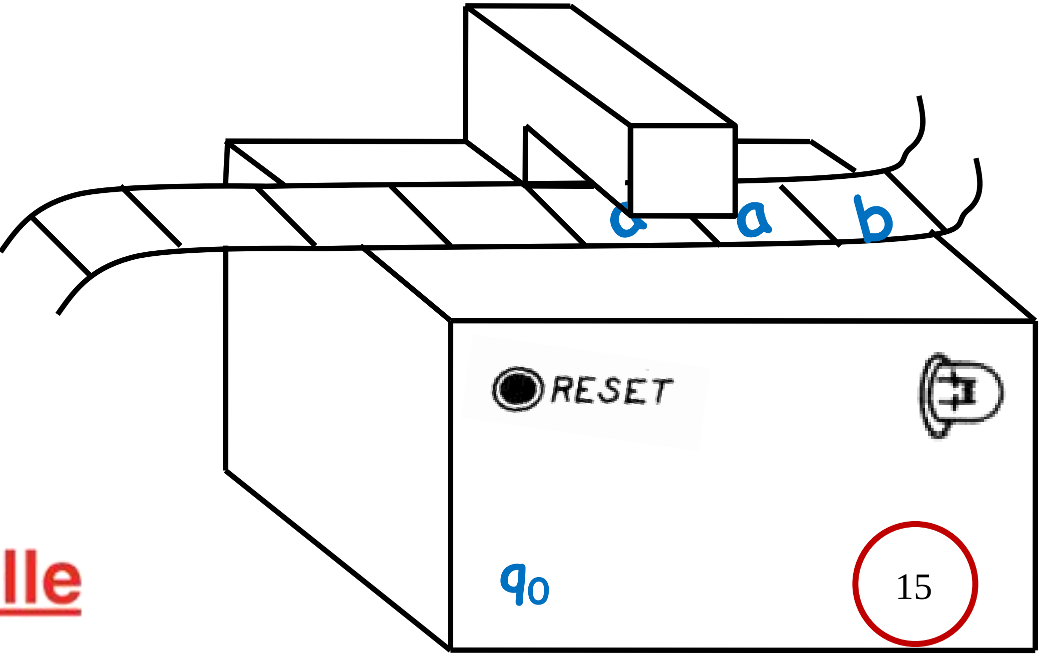


Universidad del Valle

Lenguajes regulares

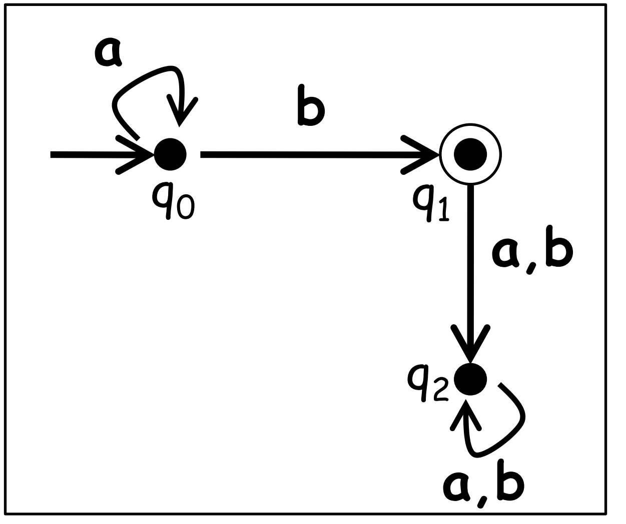
Realice el seguimiento del cómputo para la cadena **aab**

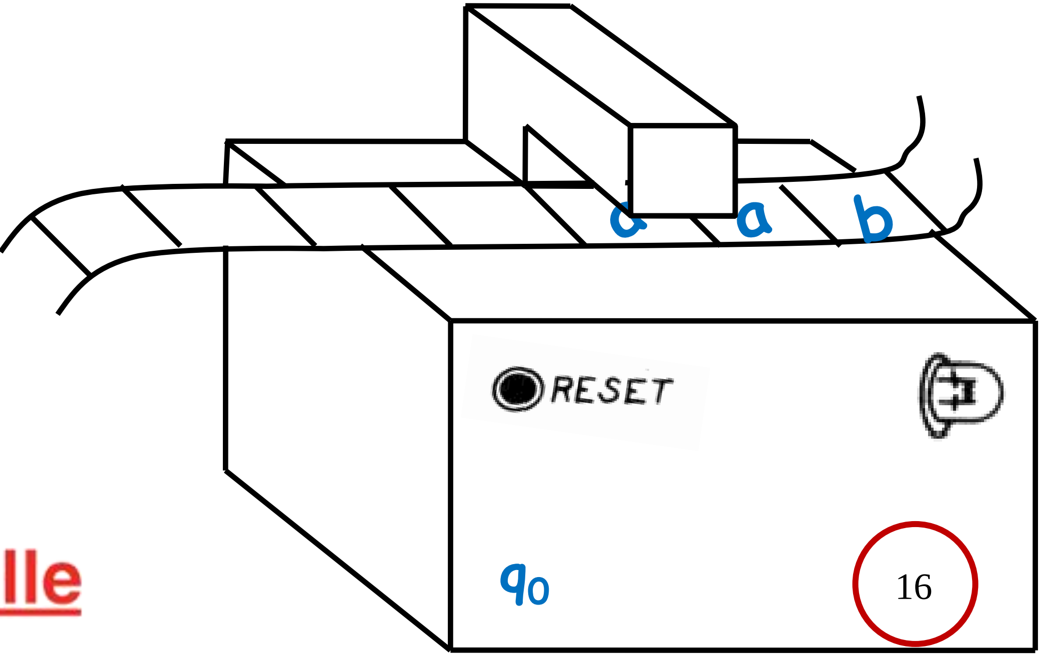




Lenguajes regulares

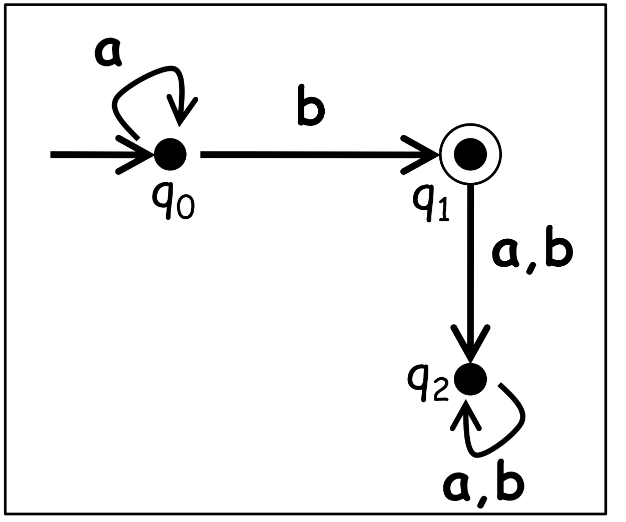
$(q_0, a) \rightarrow q_0$

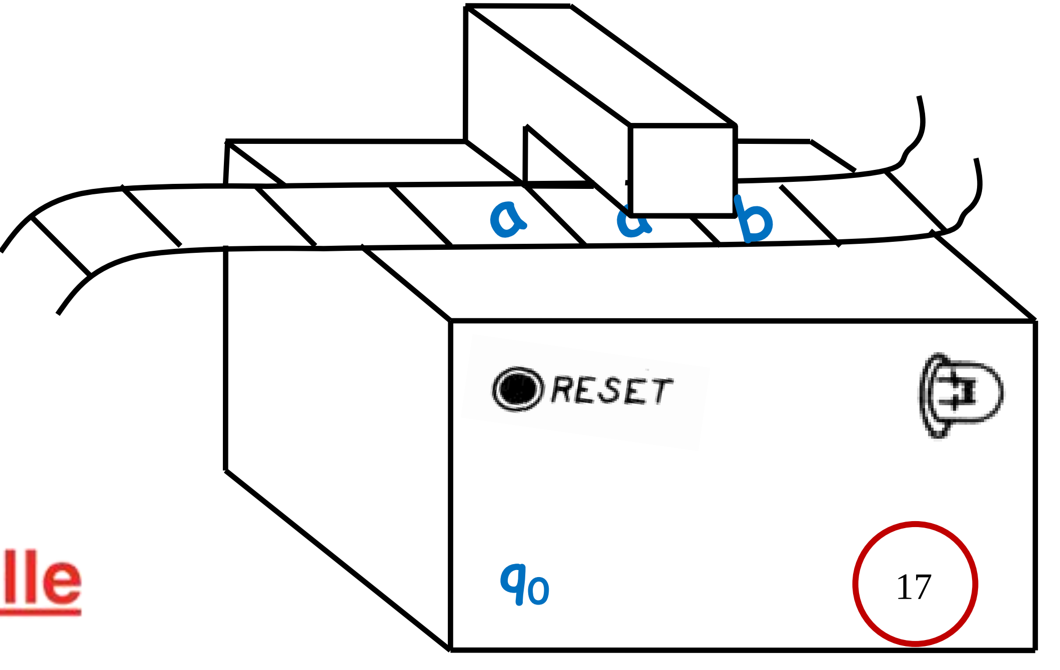




Lenguajes regulares

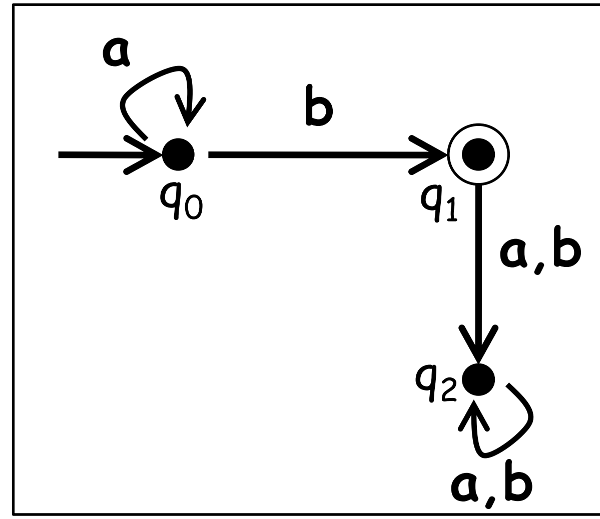
$(q_0, a) \rightarrow q_0$

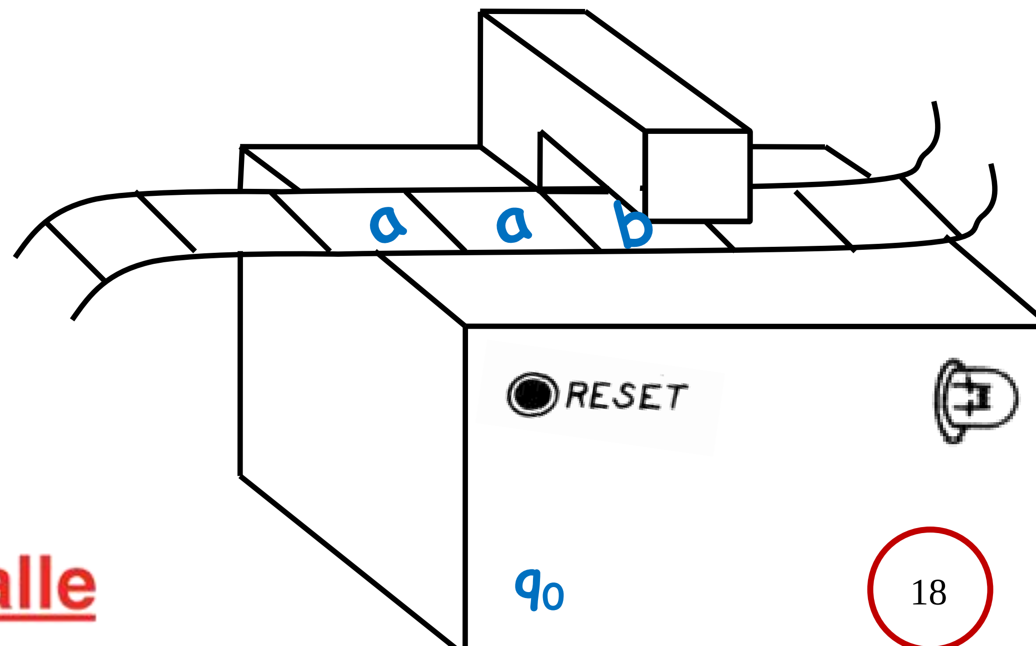




Lenguajes regulares

$(q_0,a) \rightarrow q_0$

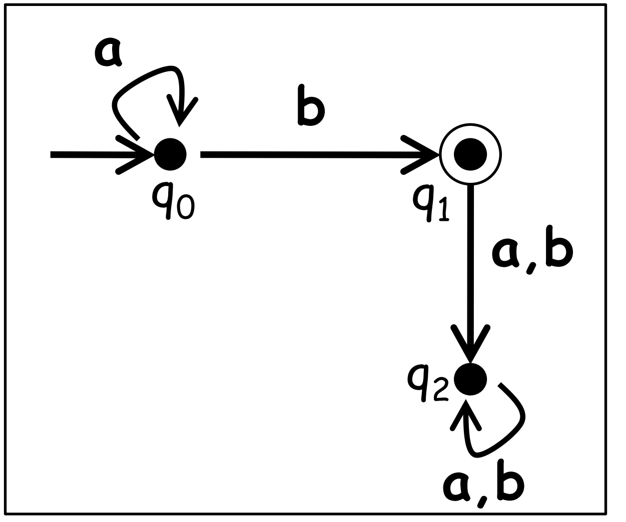


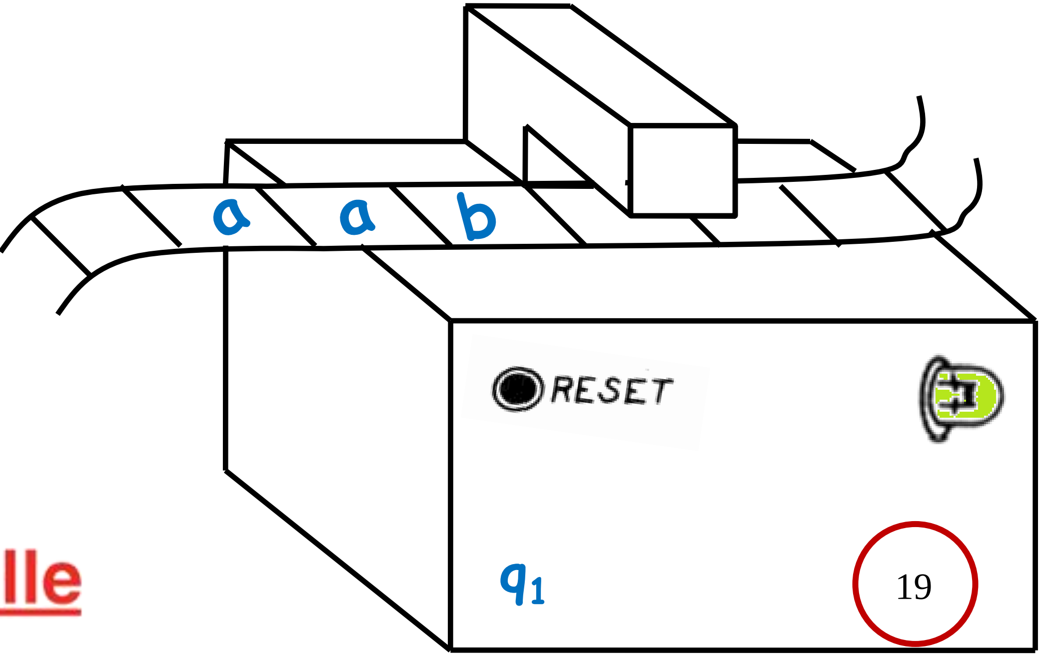


Universidad del Valle

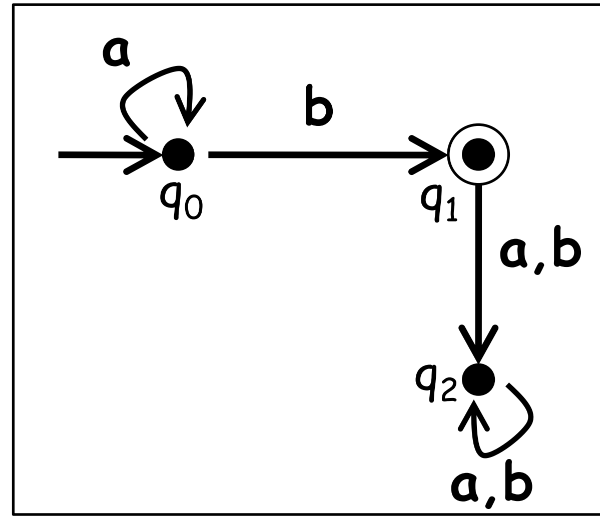
Lenguajes regulares

$(q_0,b) \rightarrow q_1$

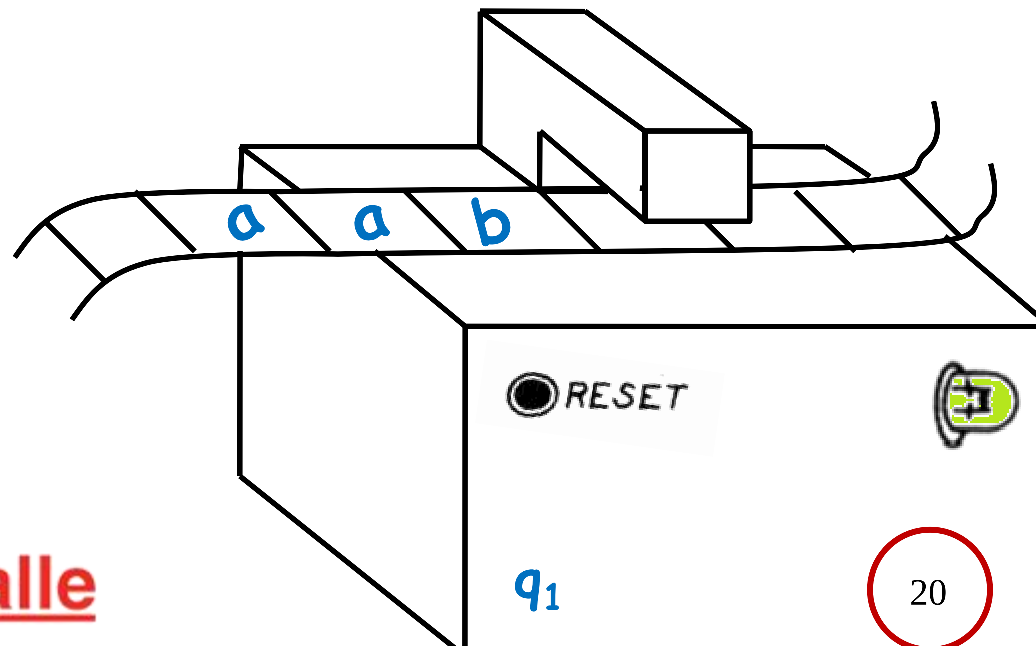




$(q_0,b) \rightarrow q_1$

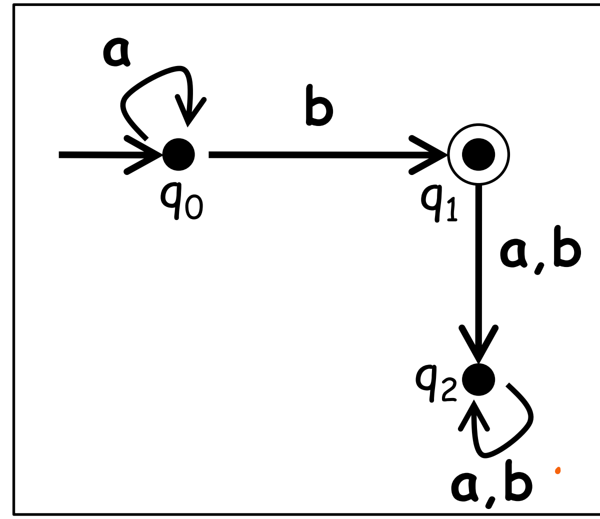


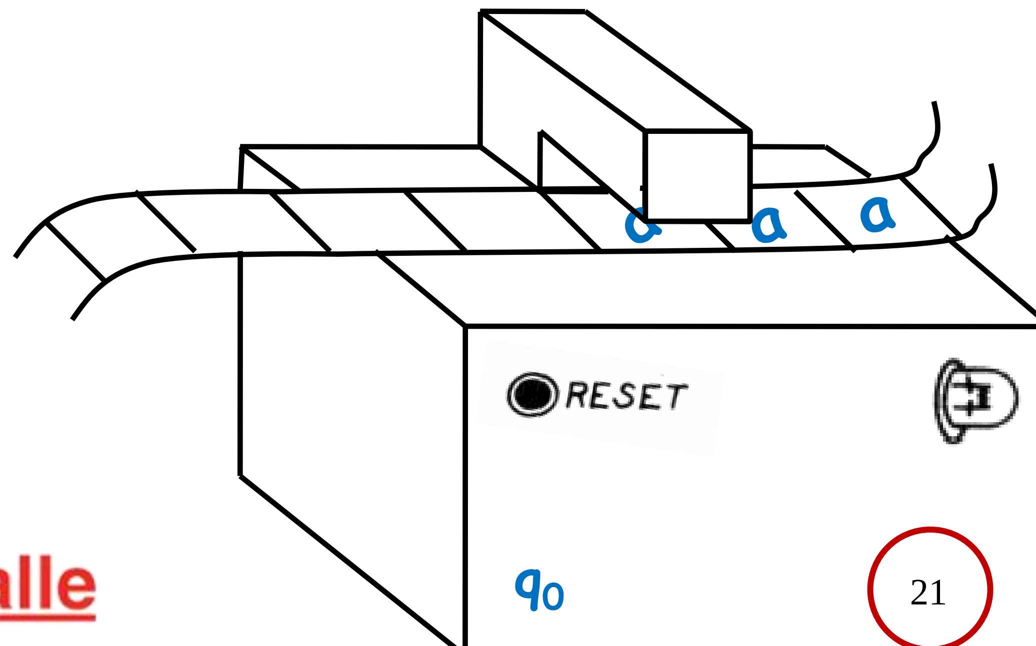
Como se consumen los símbolos en la cinta y q_1 es un estado de aceptación, se dice que el autómata reconoce **aab**



Lenguajes regulares

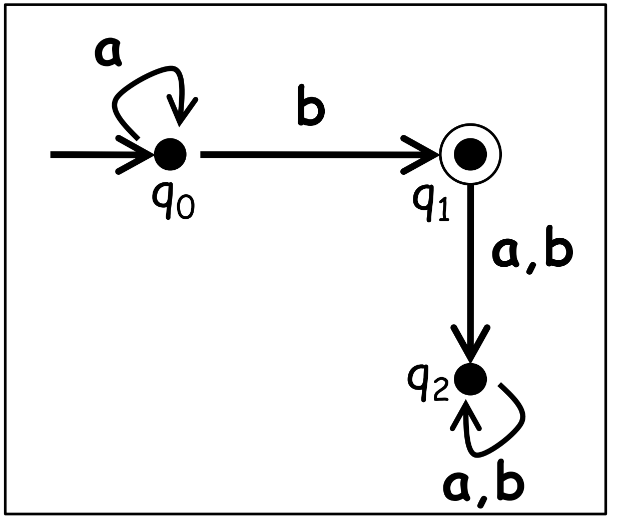
Indique si se acepta la cadena **aaa**

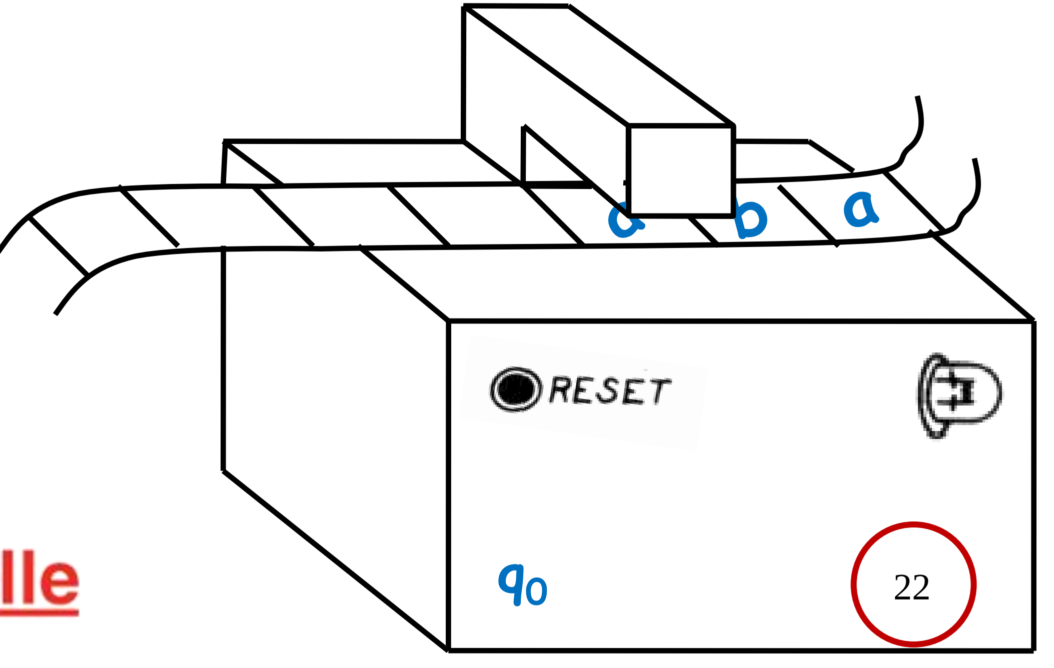




Universidad del Valle

Lenguajes regulares

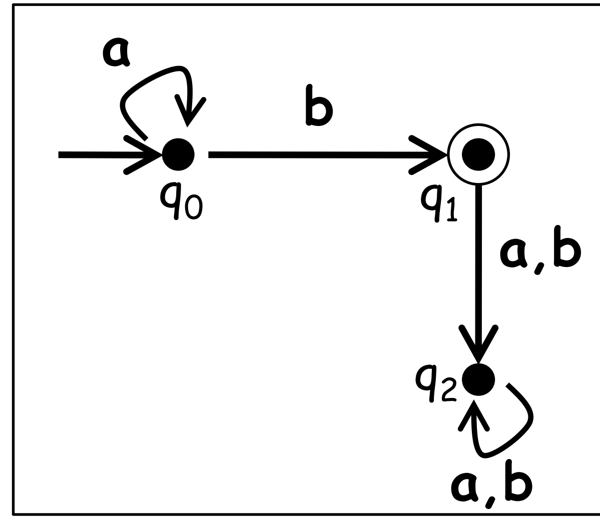


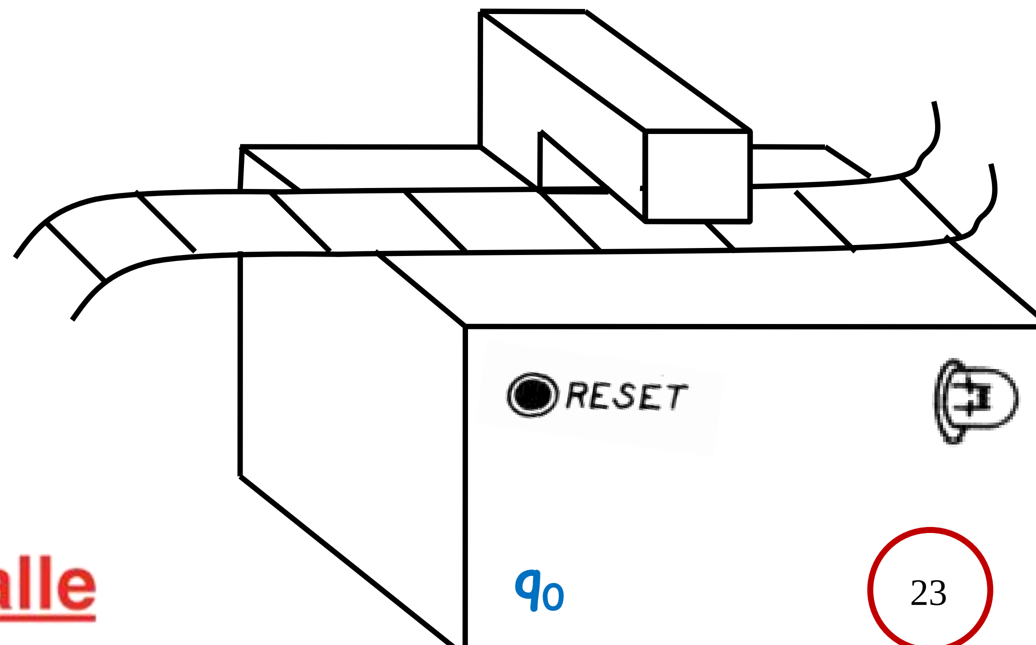


Universidad del Valle

Lenguajes regulares

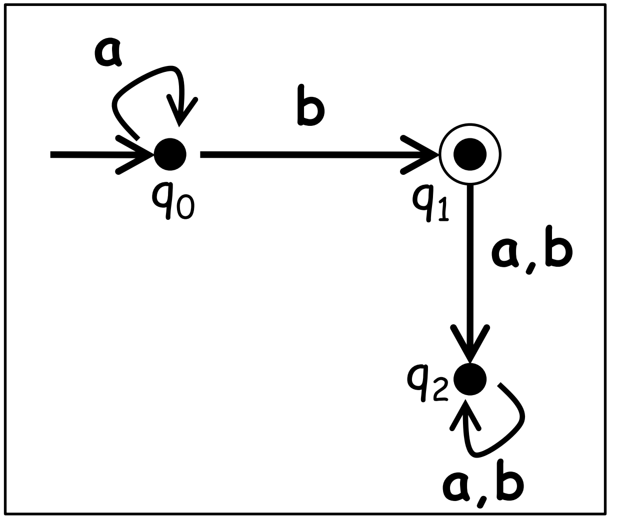
- Indique una expresión regular para el autómata

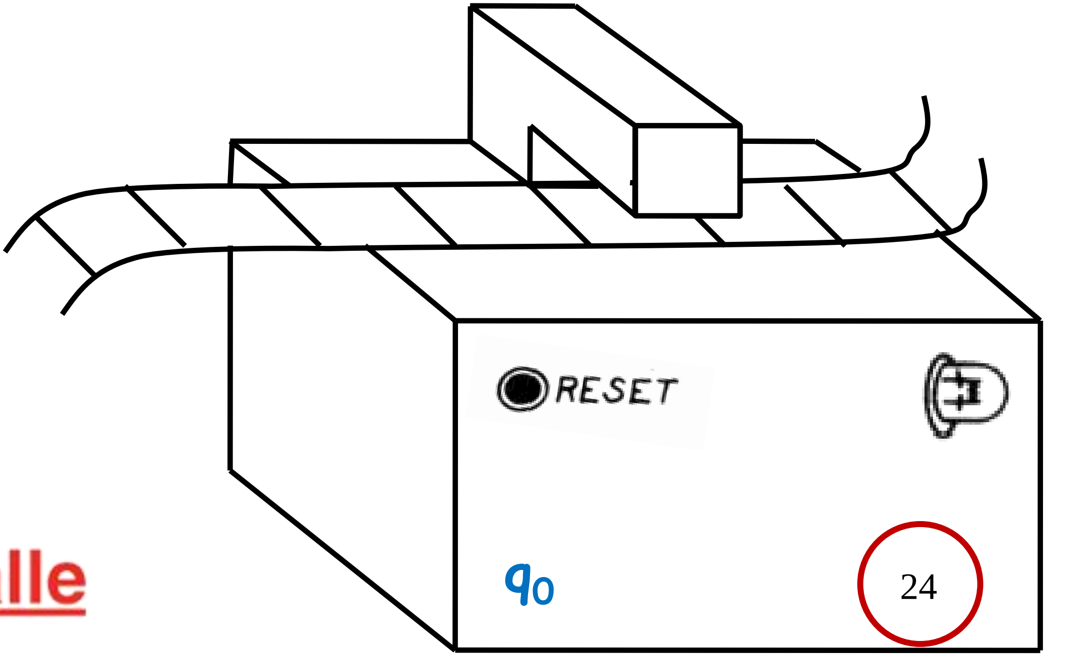




Lenguajes regulares

- El autómata acepta el lenguaje dado por **a*b**

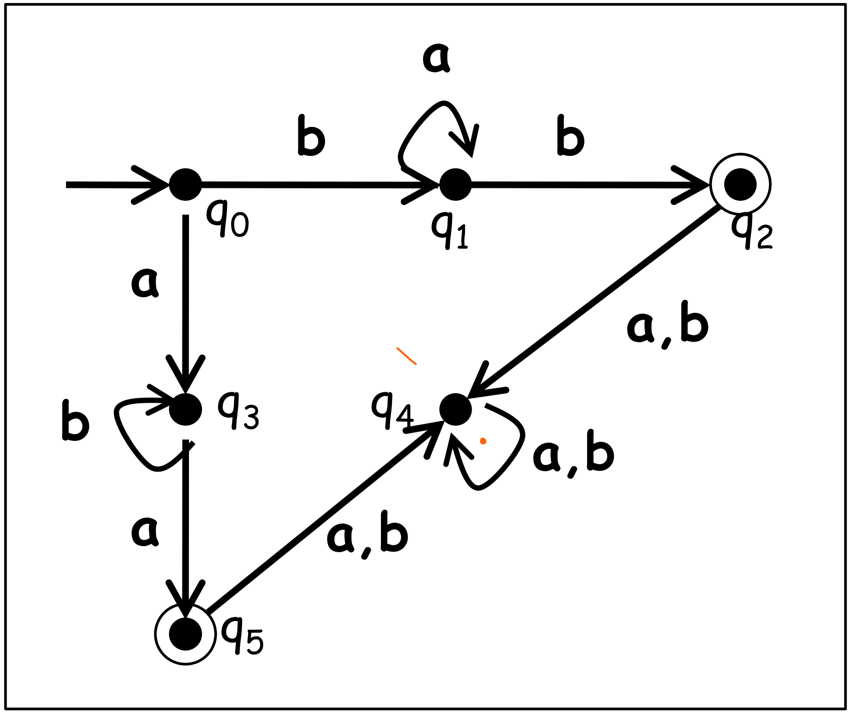




Lenguajes regulares

Indique si se aceptan las siguientes cadenas:

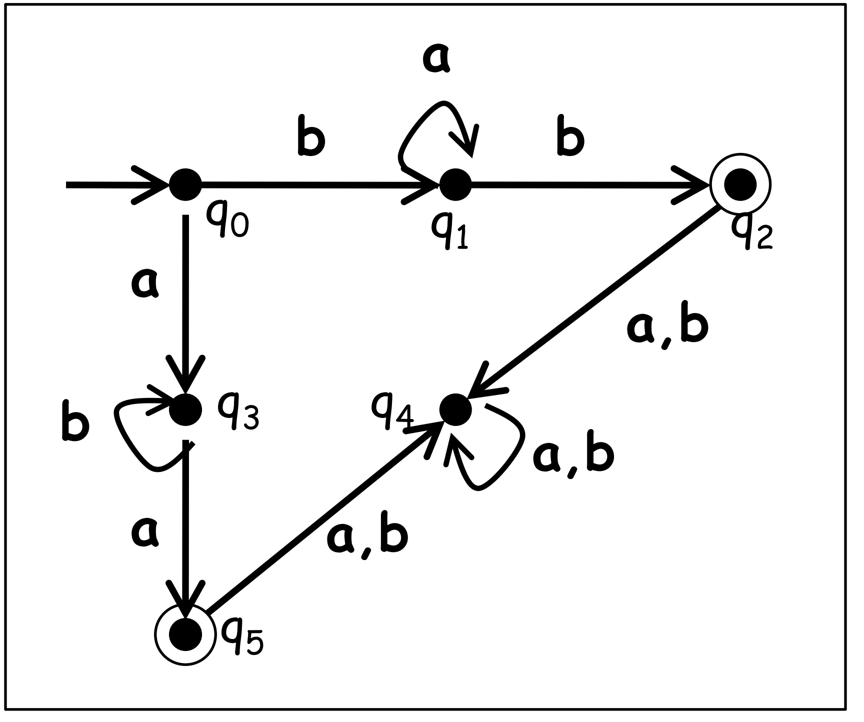
- ε
- ab
- bab
- ba^4b
- ab^3a



Lenguajes regulares

Indique si se aceptan las siguientes cadenas:

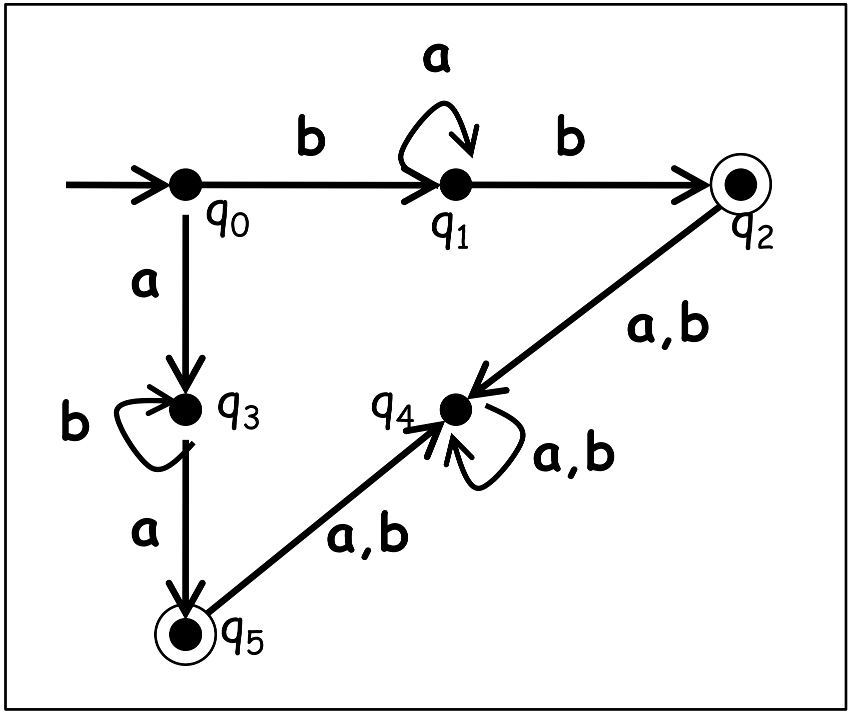
- ε
- ab
- bab
- ba^4b
- ab^3a



Lenguajes regulares

Indique si se aceptan las siguientes cadenas:

- ε
- ab

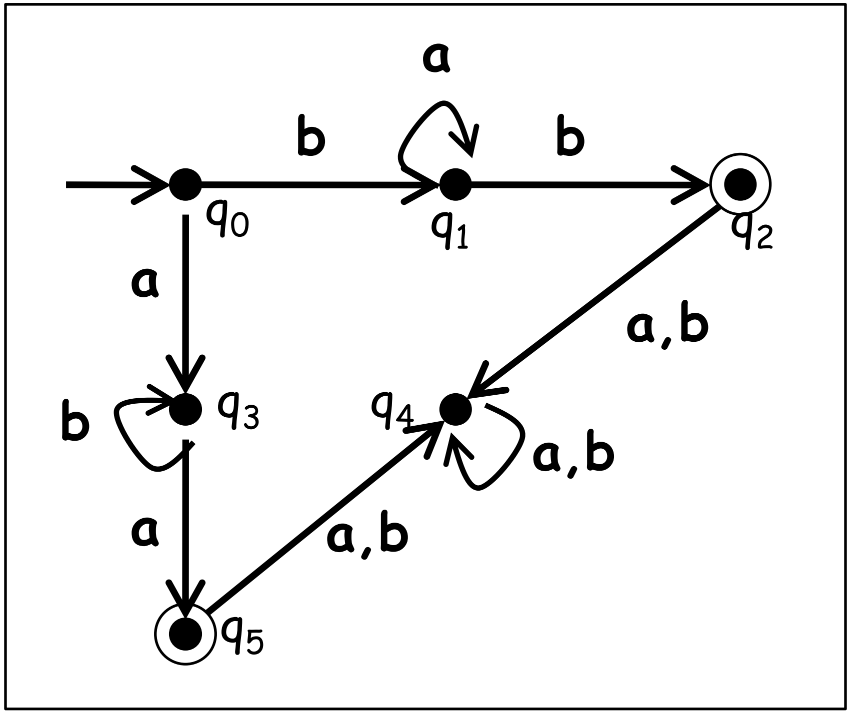


Indique un expresión regular que represente el lenguaje aceptado por el autómata

Indique si se aceptan las siguientes cadenas:

- ε
- ab
- bab
- ba^4b
- ab^3a

Indique un expresión regular que represente el lenguaje aceptado por el autómata

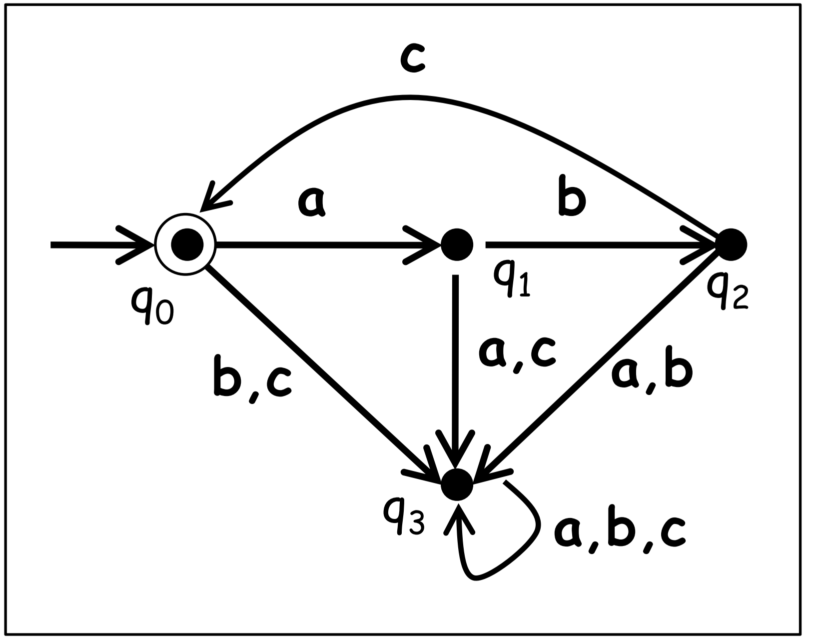


$ab^*a \cup ba^*b$

Lenguajes regulares

Indique si se aceptan las siguientes cadenas:

- ε
- abc
- $(abc)^2$
- aabc
- aba
- abca

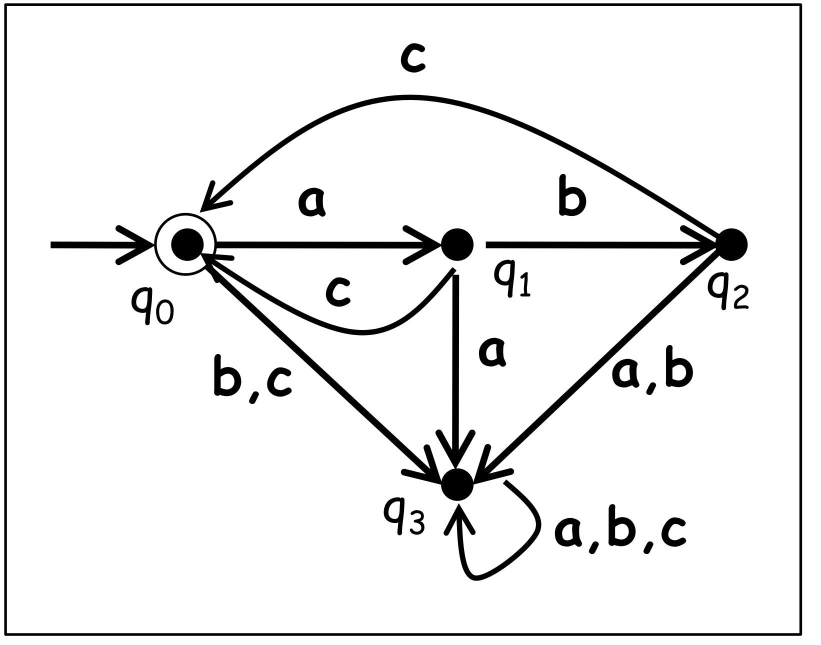


Indique un expresión regular que represente el lenguaje aceptado por el autómata

Lenguajes regulares

Indique si se aceptan las siguientes cadenas:

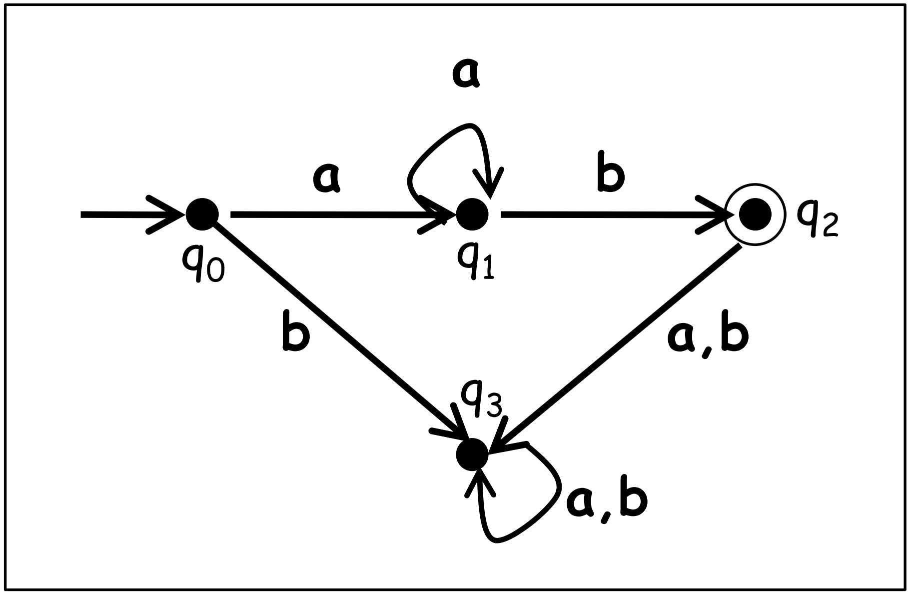
- ε
- abc
- abcac
- $(ac)^{10}$
- $a^2b^2c^2$
- $(abc)^2$
- $(abc)^2(ac)^3$



Indique un expresión regular que represente el lenguaje aceptado por el autómata

Diseñe un autómata finito que acepte $\mathbf{a^+b}$

Diseñe un autómata finito que acepte $\mathbf{a^+b}$



Expresión regular: a^+b
Lenguaje: {ab, aab, aaab, ...}

Lenguajes regulares

Diseñe un autómata finito que acepte $\mathbf{a(a \cup b)^*c}$

Lenguajes regulares

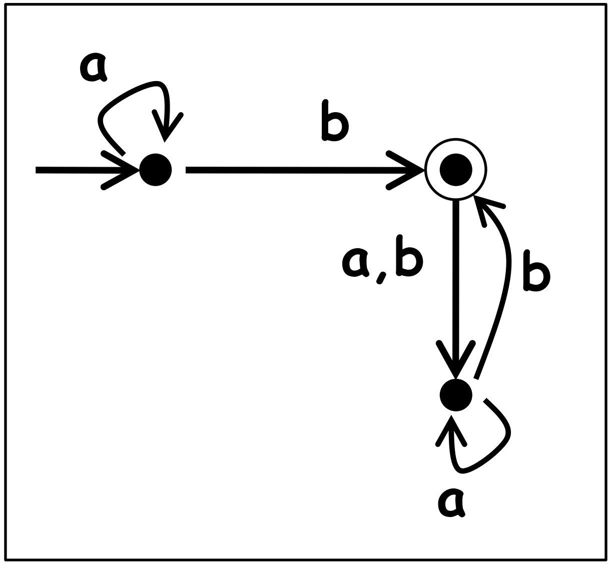
Diseñe un autómata finito que acepte **(ab)***

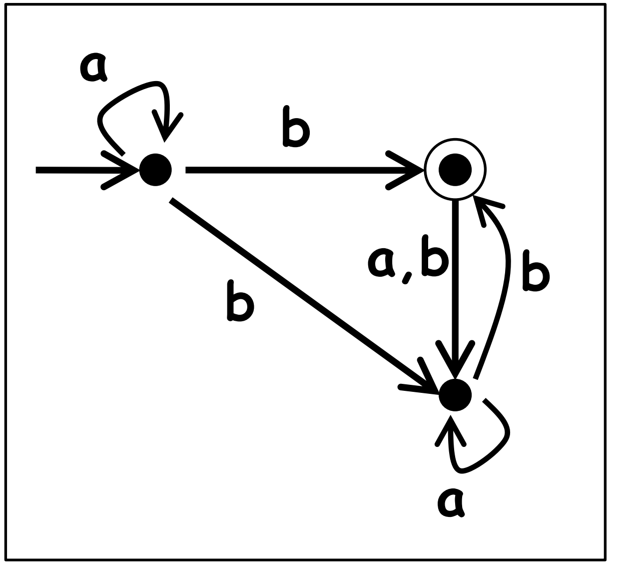
Teorema de Kleene

- Un lenguaje es regular si y sólo si es aceptado por un autómata finito

Autómatas finitos

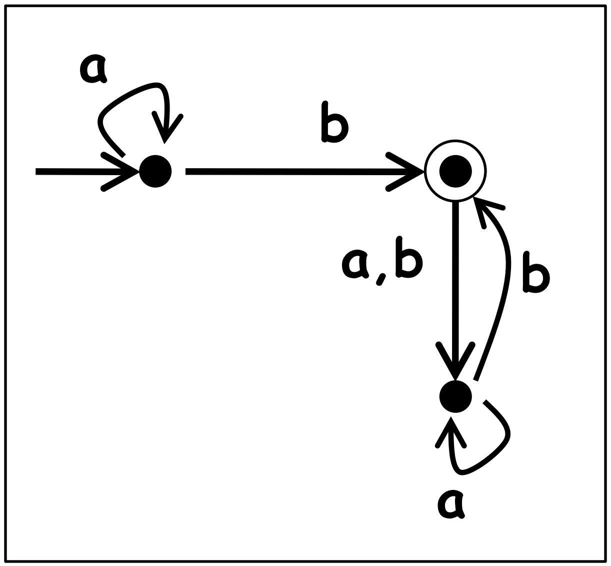
• Los autómatas finitos se dividen en autómatas finitos deterministas (**AFD**) y en no deterministas (**AFN**)



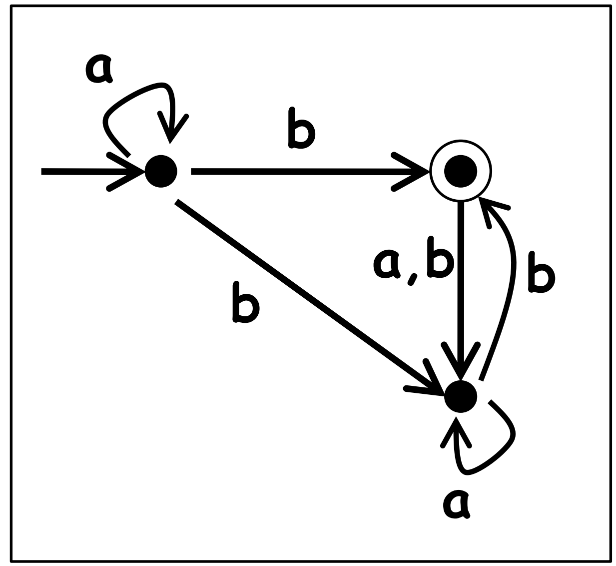


Autómatas finitos

- Los autómatas finitos se dividen en autómatas finitos deterministas (**AFD**) y en no deterministas (**AFN**)



Dado un estado **q** y un símbolo **x**, se tiene una sola arista de transición



Dado un estado **q** y un símbolo **x**, se tienen varias transiciones posibles

Autómatas finitos deterministas (AFD)

Un AFD es una colección de cinco elementos:

- Un alfabeto Σ
- Una colección finita de estados **Q**
- Un estado inicial $\mathbf{q_0}$
- Una colección finita de estados de aceptación **T**
- Una **función** δ**:Q**xΣ**→Q** que determina el único estado siguiente para el par (q_i,σ) correspondiente al estado actual q_i y la entrada σ

Autómatas finitos deterministas (AFD)

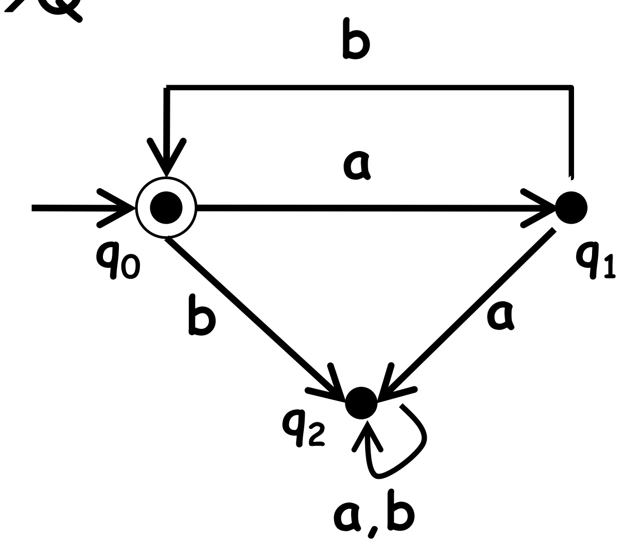
Un AFD es una colección de cinco elementos:

- Un alfabeto Σ
- Una colección finita de estados **Q**
- Un estado inicial $\mathbf{q_0}$
- Una colección finita de estados de aceptación **T**
- Una función δ:**Q**$\times\Sigma\rightarrow$**Q** que determina el único estado siguiente para el par (q_i,σ) correspondiente al estado actual q_i y la entrada σ

δ debe ser una **función** para que exista el determinismo

Autómatas finitos deterministas (AFD)

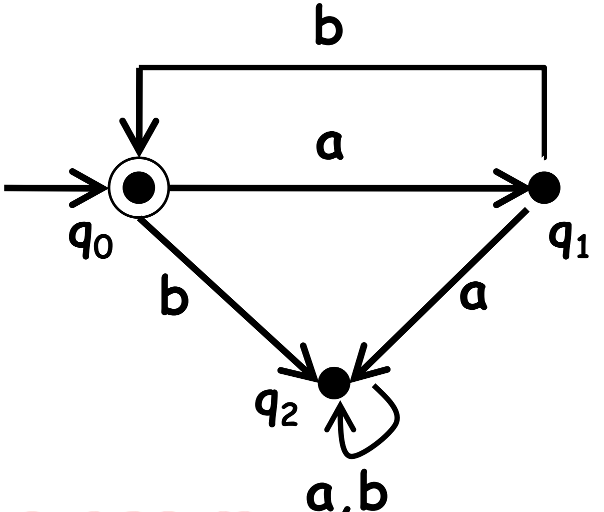
- Un alfabeto Σ
- Una colección finita de estados **Q**
- Un estado inicial $\mathbf{q_0}$
- Una colección finita de estados de aceptación **T**
- Una función δ:**Q**x$\Sigma\rightarrow$**Q**



Autómatas finitos deterministas (AFD)

- Σ
- Q
- Estado inicial
- T
- $\delta: Q \times \Sigma \to Q$

δ	a	b
q_0		
q_1		
q_2		

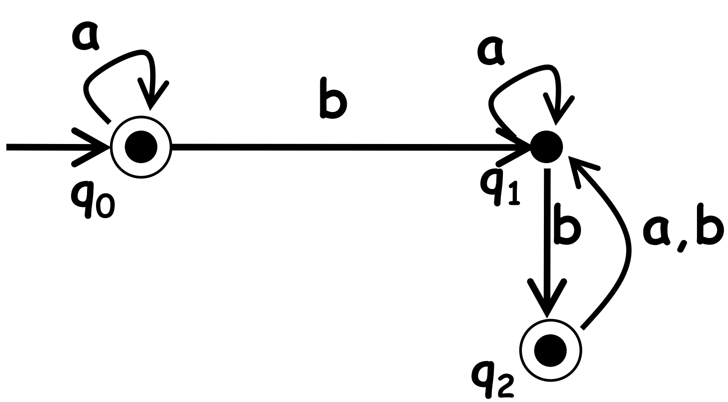


Lenguajes regulares

Indique los 5 elementos que definen el siguiente autómata:

- Σ
- Q
- Estado inicial
- T
- $\delta: Q \times \Sigma \rightarrow Q$

δ	a	b
q_0		
q_1		
q_2		



Muestre el diagrama de transición para el autómata:

- $\Sigma=\{a,b\}$
- $Q=\{q_0,q_1\}$
- Estado inicial q_0
- $T=\{q_0\}$
- $\delta:Q\times\Sigma\rightarrow Q$

δ	**a**	**b**
$\mathbf{q_0}$	q_0	q_1
$\mathbf{q_1}$	q_1	q_1

Indique el lenguaje aceptado

Muestre el diagrama de transición para el autómata:

- $\Sigma=\{a,b\}$
- $Q=\{q_0,q_1,q_2\}$
- Estado inicial q_0
- $T=\{q_1\}$
- $\delta: Q \times \Sigma \rightarrow Q$

δ	**a**	**b**
q_0	q_1	q_2
q_1	q_1	q_2
q_2	q_2	q_2

Indique el lenguaje aceptado

Lenguajes regulares

Muestre el diagrama de transición para el autómata:

- $\Sigma=\{a,b\}$
- $Q=\{q_0,q_1,q_2,q_3\}$
- Estado inicial q_0
- $T=\{q_2\}$
- $\delta:Q\times\Sigma\rightarrow Q$

δ	**a**	**b**
q_0	q_1	q_3
q_1	q_1	q_2
q_2	q_3	q_2
q_3	q_3	q_3

Indique el lenguaje aceptado

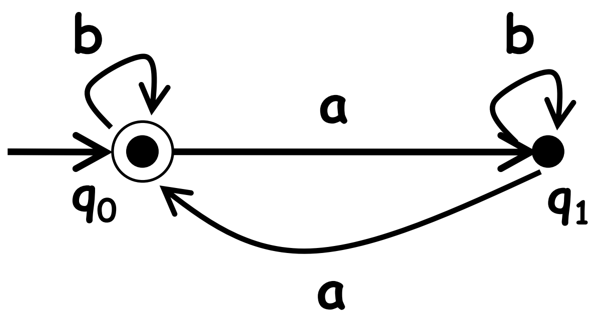
Diseñe un AFD sobre $\Sigma=\{a,b\}$ que reconozca el lenguaje de todas las palabras que contienen un número par de a's. Se aceptan cadenas que tienen cero a's

- Muestre el diagrama de transición
- Exprese el autómata formalmente
- Indique la expresión regular

Lenguajes regulares

- $\Sigma=\{a,b\}$
- $Q=\{q_0,q_1\}$
- Estado inicial q_0
- $T=\{q_0\}$
- $\delta: Q \times \Sigma \rightarrow Q$

δ	a	b
q_0	q_1	q_0
q_1	q_0	q_1



El autómata acepta: $(b^* \cup (ab^*a)^*)^*$

Lenguajes regulares

Diseñe un AFD sobre $\Sigma=\{a,b\}$ que reconozca b^*a^+

- Muestre el diagrama de transición
- Exprese el autómata formalmente

Lenguajes regulares

Diseñe un AFD sobre $\Sigma=\{a,b\}$ que reconozca el lenguaje de todas las palabras que tienen al menos una a

- Muestre el diagrama de transición
- Indique la expresión regular

Diseñe un AFD sobre $\Sigma=\{a,b\}$ que reconozca a^+b^+

- Muestre el diagrama de transición
- Indique la expresión regular

Lenguajes regulares

Diseñe un AFD sobre $\Sigma=\{a,b\}$ que reconozca el lenguaje de todas las cadenas que tienen un número par de símbolos (incluida la cadena vacía)

- Muestre el diagrama de transición
- Exprese el autómata formalmente
- Indique la expresión regular

Lenguajes regulares

Diseñe un AFD sobre $\Sigma=\{a,b\}$ que reconozca el lenguaje de todas las cadenas que tienen un número par de símbolos (sin incluir la cadena vacía)

- Muestre el diagrama de transición
- Exprese el autómata formalmente
- Indique la expresión regular

Lenguajes regulares

Construya el autómata que reconozca el lenguaje

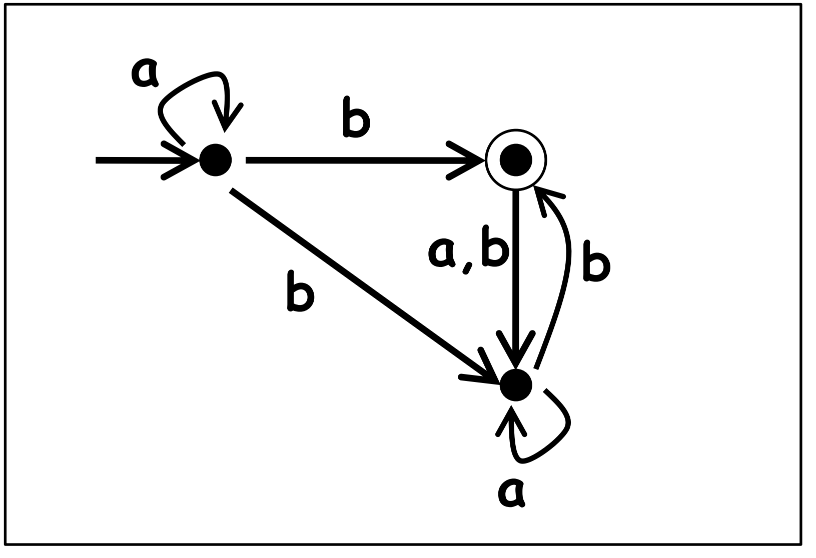
$(ab \cup aa(ba)^*bb)^*$

Autómatas finitos no deterministas (AFN)

• Si se permite que desde algún estado se realicen cero, dos o más transiciones mediante el mismo símbolo de entrada, se dice que el **autómata finito es no determinista**

Autómatas finitos no deterministas (AFN)

- Si se permite que desde algún estado se realicen cero, dos o más transiciones mediante el mismo símbolo de entrada, se dice que el **autómata finito es no determinista**



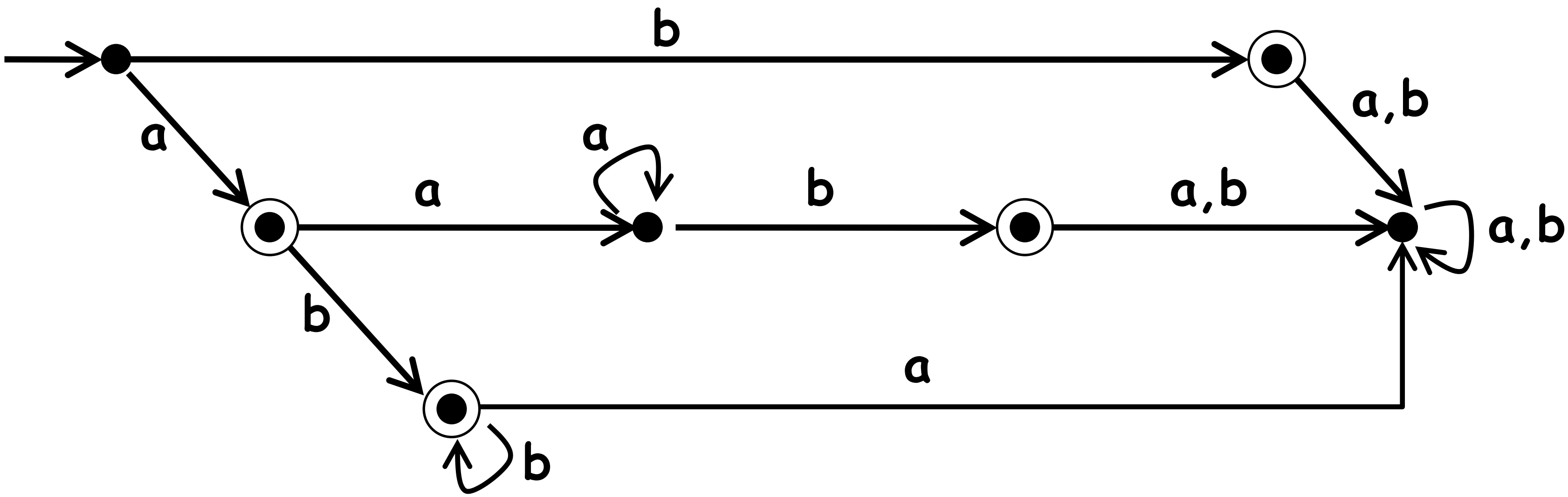
Autómatas finitos no deterministas (AFN)

• Los AFN se utilizan porque pueden ser más simples que los AFD

Construya el AFD que acepta $a^*b \cup ab^*$

Autómatas finitos no deterministas (AFN)

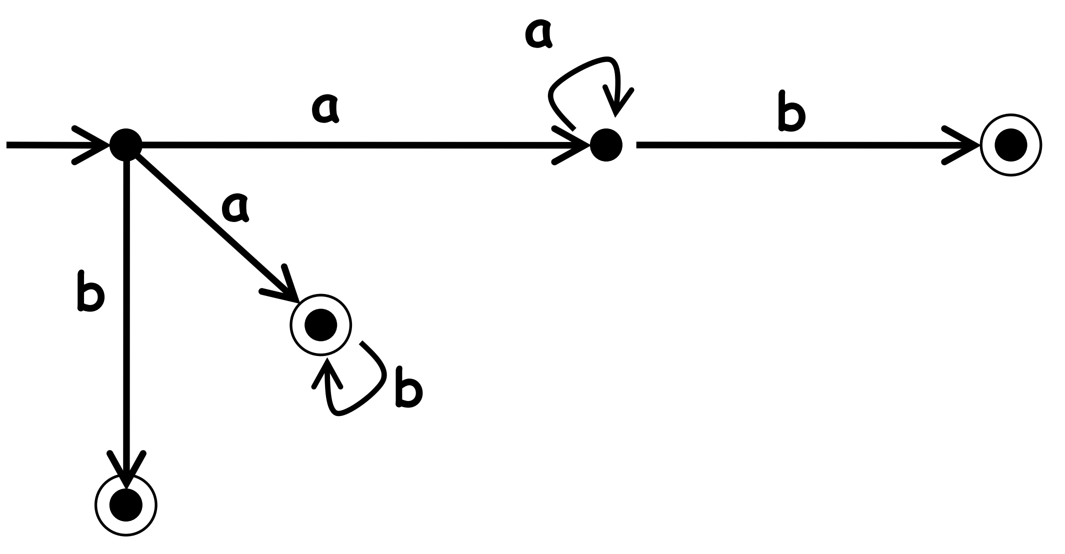
• Los AFN se utilizan porque pueden ser más simples que los AFD



AFD que acepta a*b∪ab*

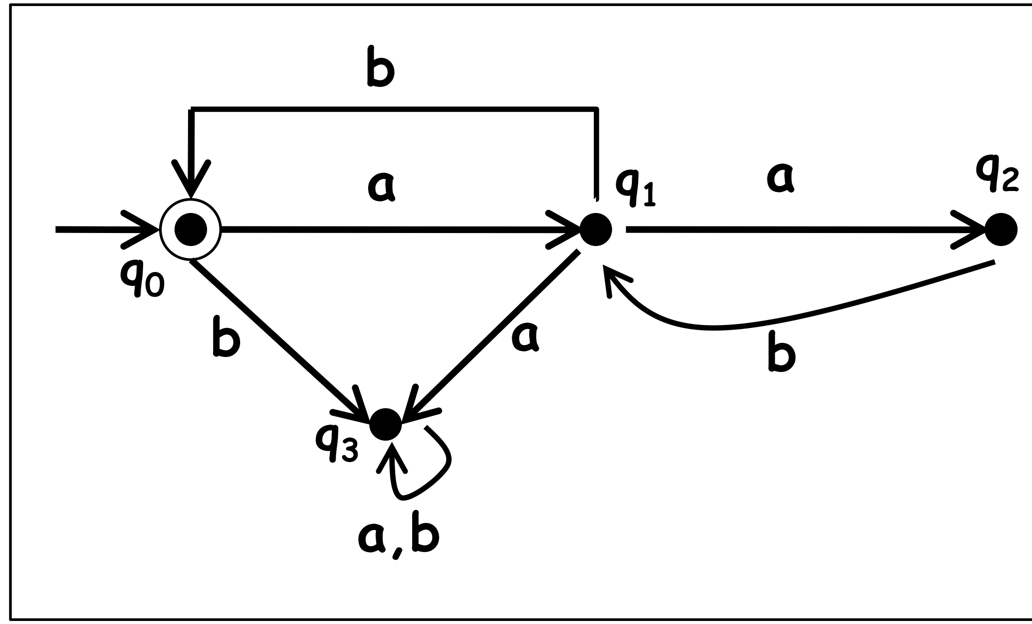
Autómatas finitos no deterministas (AFN)

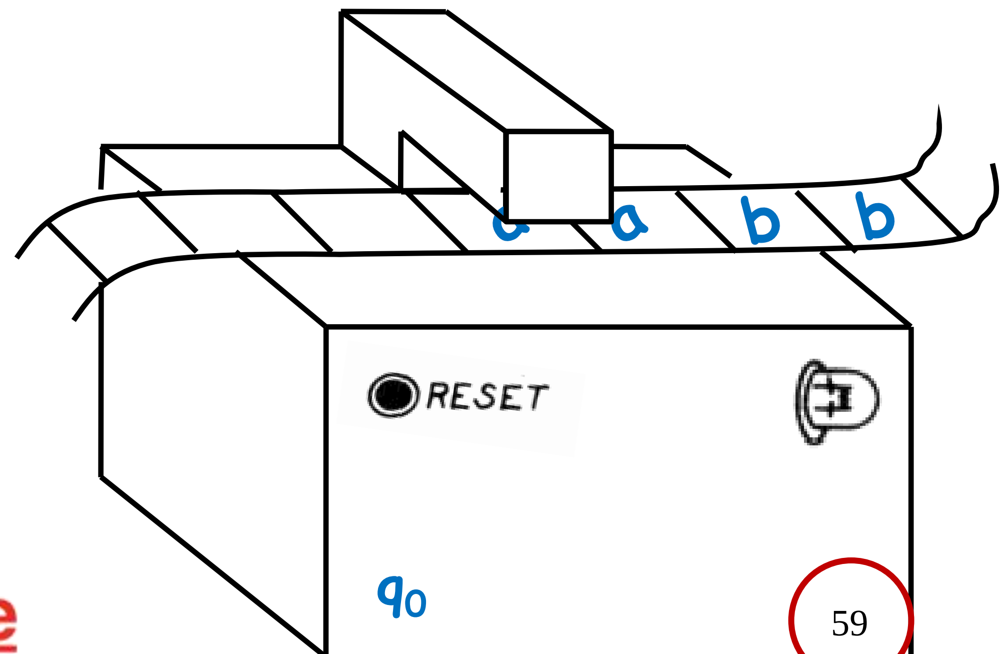
- Los AFN se utilizan porque pueden ser más simples que los AFD



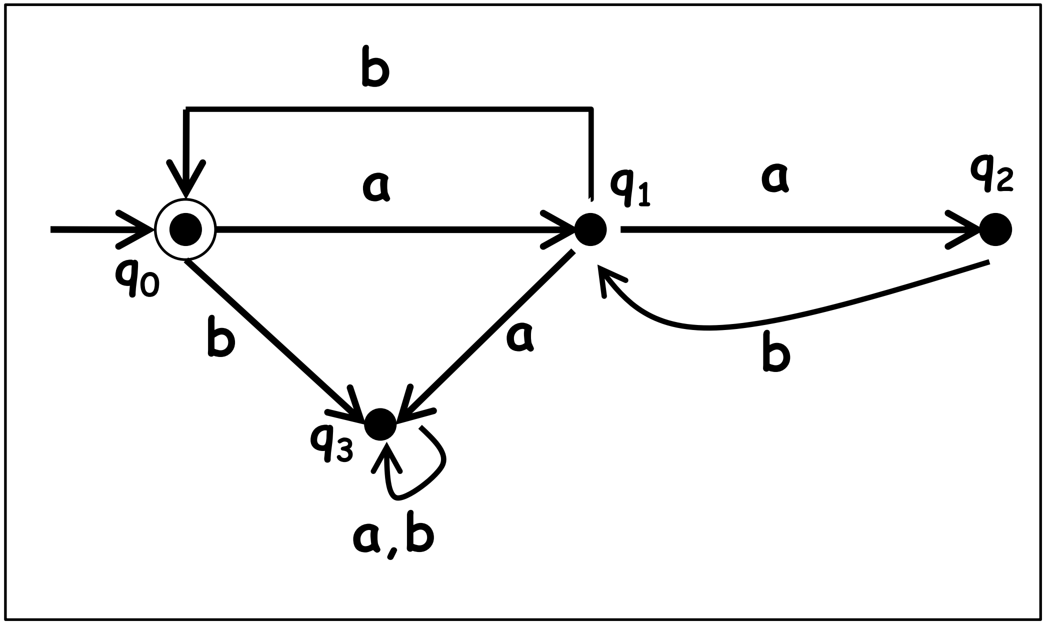
AFN que acepta a*b∪ab*

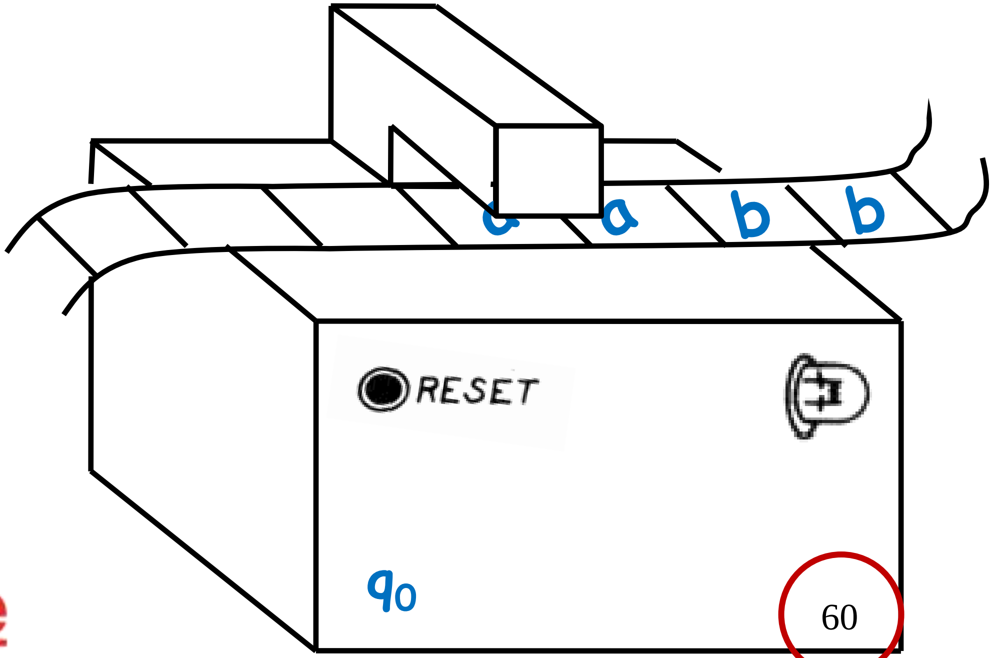
- ¿El autómata finito acepta o rechaza la cadena **aabb**?





- En un AFN se puede suponer que si existe un recorrido en el diagrama de transición que termine en un estado de aceptación, el autómata lo encuentra





Universidad del Valle

Autómatas finitos no deterministas (AFN)

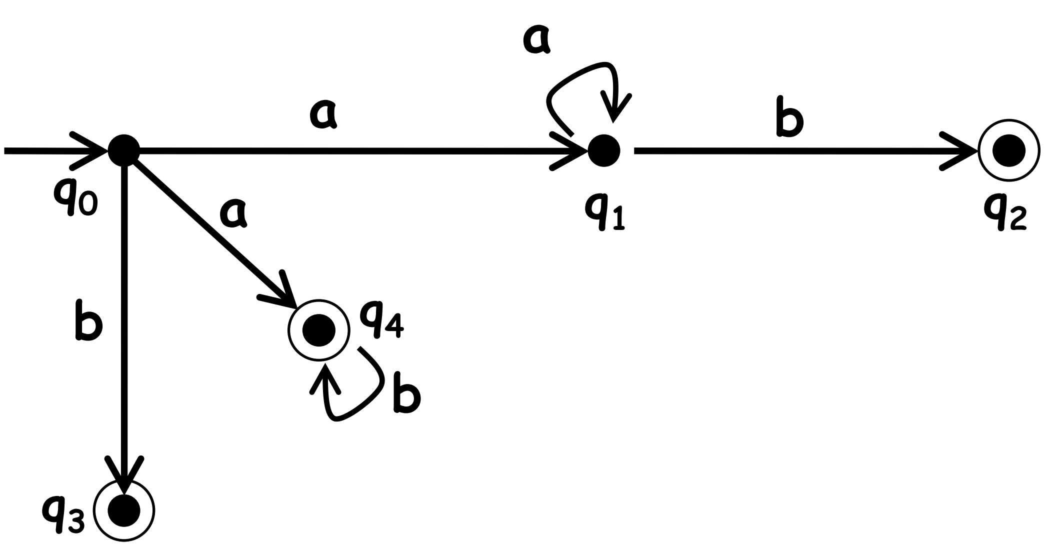
Un AFN es una colección de cinco elementos:

- Un alfabeto Σ
- Una colección finita de estados **Q**
- Un estado inicial $\mathbf{q_0}$
- Una colección finita de estados de aceptación **T**
- Una **relación** Δ sobre $(\mathbf{Q} \times \Sigma) \rightarrow \mathbf{2^Q}$ denominada **relación de transición.** 2^Q es el conjunto potencia de Q (subconjuntos de Q)

Lenguajes regulares

- $\Sigma=\{a,b\}$
- $Q=\{q_0,q_1,q_2,q_3,q_4\}$
- Estado inicial q_0
- $T=\{q_2,q_3,q_4\}$
- $\Delta:Q\times\Sigma\rightarrow 2^Q$

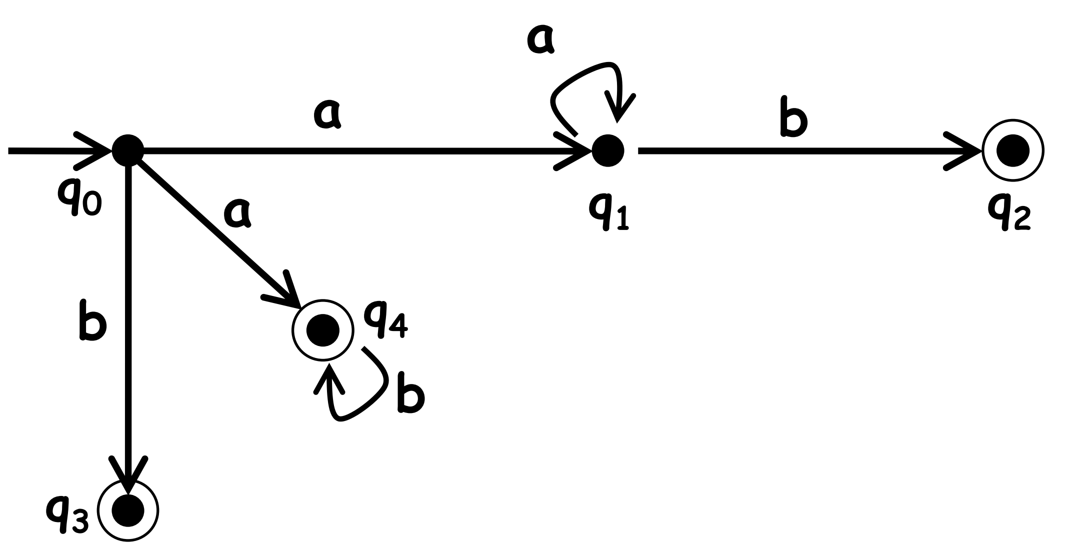
Δ	a	b
q_0		
q_1		
q_2		
q_3		
q_4		



Lenguajes regulares

- $\Sigma=\{a,b\}$
- $Q=\{q_0,q_1,q_2,q_3,q_4\}$
- Estado inicial q_0
- $T=\{q_2,q_3,q_4\}$
- $\Delta : Q \times \Sigma \rightarrow 2^Q$

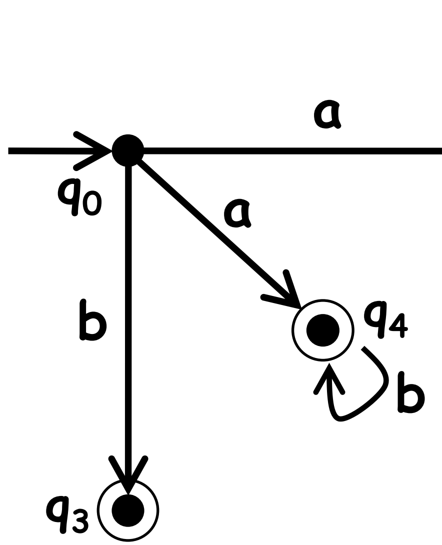
Δ	a	b
q_0	$\{q_1, q_4\}$	$\{q_3\}$
q_1	$\{q_1\}$	$\{q_2\}$
q_2	$\varnothing$	$\varnothing$
q_3	$\varnothing$	$\varnothing$
q_4	$\varnothing$	$\{q_4\}$



Lenguajes regulares

- $\Sigma=\{a,b\}$
- $Q=\{q_0,q_1,q_2,q_3,q_4\}$
- Estado inicial q_0
- $T=\{q_2,q_3,q_4\}$
- $\Delta:Q\times\Sigma\rightarrow 2^Q$

Δ	a	b
q_0	$\{q_1, q_4\}$	$\{q_3\}$
q_1	$\{q_1\}$	$\{q_2\}$
q_2	$\varnothing$	$\varnothing$
q_3	$\varnothing$	$\varnothing$
q_4	$\varnothing$	$\{q_4\}$

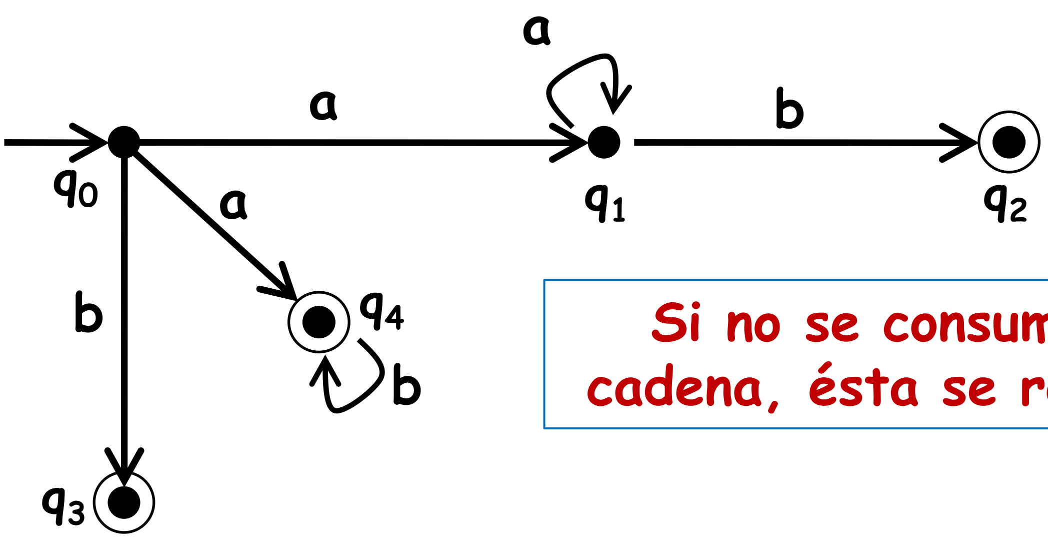


¿La cadena **baa** se acepta o rechaza por el autómata?

Lenguajes regulares

- $\Sigma=\{a,b\}$
- $Q=\{q_0,q_1,q_2,q_3,q_4\}$
- Estado inicial q_0
- $T=\{q_2,q_3,q_4\}$
- $\Delta:Q\times\Sigma\rightarrow 2^Q$

Δ	a	b
q_0	$\{q_1, q_4\}$	$\{q_3\}$
q_1	$\{q_1\}$	$\{q_2\}$
q_2	$\varnothing$	$\varnothing$
q_3	$\varnothing$	$\varnothing$
q_4	$\varnothing$	$\{q_4\}$



Si no se consume la cadena, ésta se rechaza

Universidad del Valle

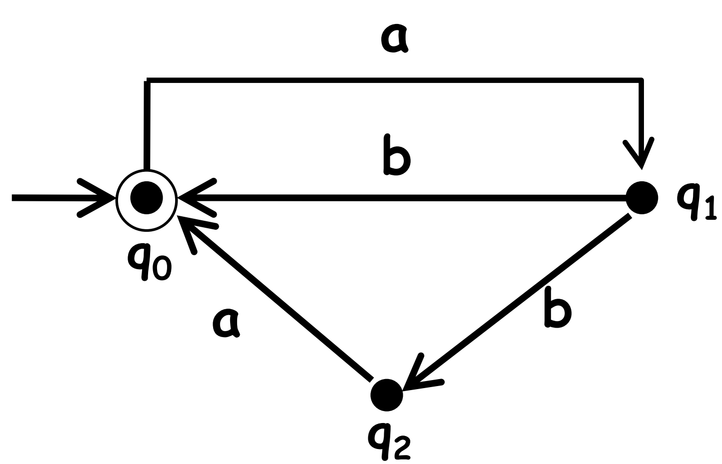
Lenguajes regulares

Represente formalmente el AFN

- Σ
- Q
- Estado inicial
- T
- Δ

Δ	a	b
q_0		
q_1		
q_2		

Indique el lenguaje aceptado



Diseñe el AFN que se especifica a continuación:

- $\Sigma=\{a,b\}$
- $Q=\{q_0,q_1,q_2\}$
- Estado inicial q_0
- $T=\{q_2\}$
- $\Delta: Q \times \Sigma \rightarrow 2^Q$

Δ	**a**	**b**
q_0	$\{q_0, q_1\}$	$\{q_0\}$
q_1	$\varnothing$	$\{q_2\}$
q_2	$\varnothing$	$\varnothing$

Indique el lenguaje aceptado

Diseñe el AFN que se especifica a continuación:

- $\Sigma=\{a,b\}$
- $Q=\{q_0,q_1,q_2,q_3,q_4\}$
- Estado inicial q_0
- $T=\{q_2,q_4\}$
- $\Delta: Q \times \Sigma \rightarrow 2^Q$

Δ	a	b
q_0	$\{q_0, q_3\}$	$\{q_0, q_1\}$
q_1	$\varnothing$	$\{q_2\}$
q_2	$\{q_2\}$	$\{q_2\}$
q_3	$\{q_4\}$	$\varnothing$
q_4	$\{q_4\}$	$\{q_4\}$

Indique el lenguaje aceptado

Lenguajes regulares

Diseñe un AFN sobre $\Sigma=\{a,b\}$ que reconozca el lenguaje de todas las cadenas que terminan en b dado por la expresión regular $(a\cup b)^*b$

- Muestre el diagrama de transición
- Exprese el autómata formalmente

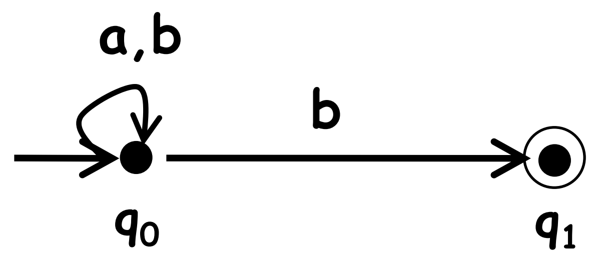
Lenguajes regulares

Diseñe un AFN sobre $\Sigma=\{a,b\}$ que reconozca el lenguaje de todas las cadenas que tienen al menos dos a's consecutivas dado por la expresión regular $(a\cup b)^*aa(a\cup b)^*$

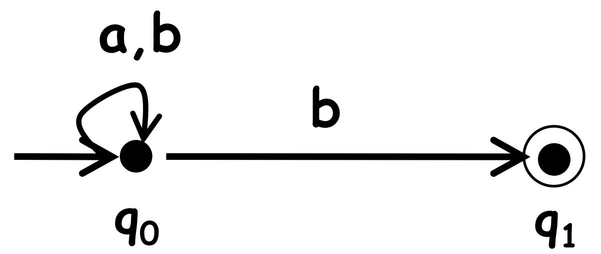
- Muestre el diagrama de transición
- Exprese el autómata formalmente

Equivalencia entre AFD y AFN

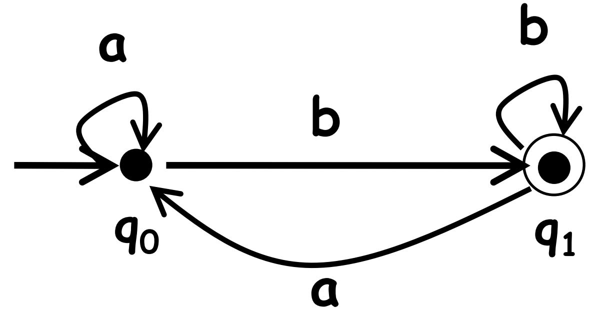
Considere el AFN que reconoce el lenguaje de las palabras sobre $\Sigma=\{a,b\}$ que terminan en b



AFN que reconoce el lenguaje de las palabras sobre $\Sigma=\{a,b\}$ que terminan en b



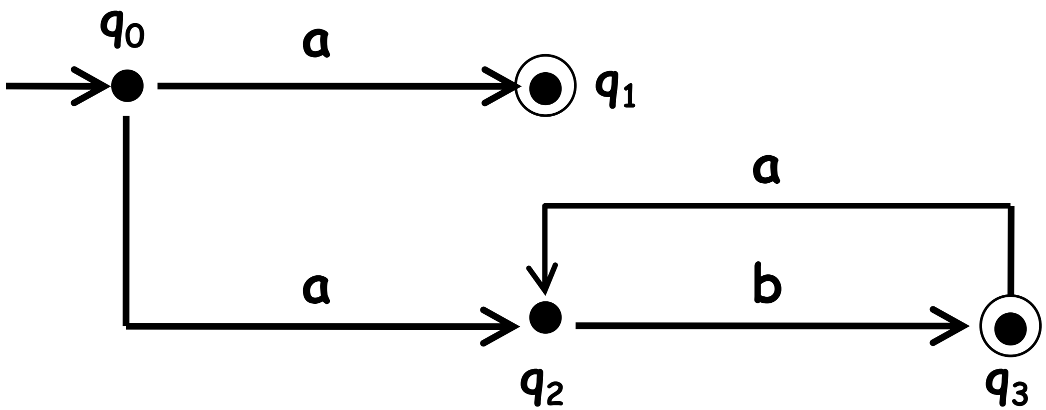
AFD que reconoce el mismo lenguaje



Equivalencia entre AFD y AFN

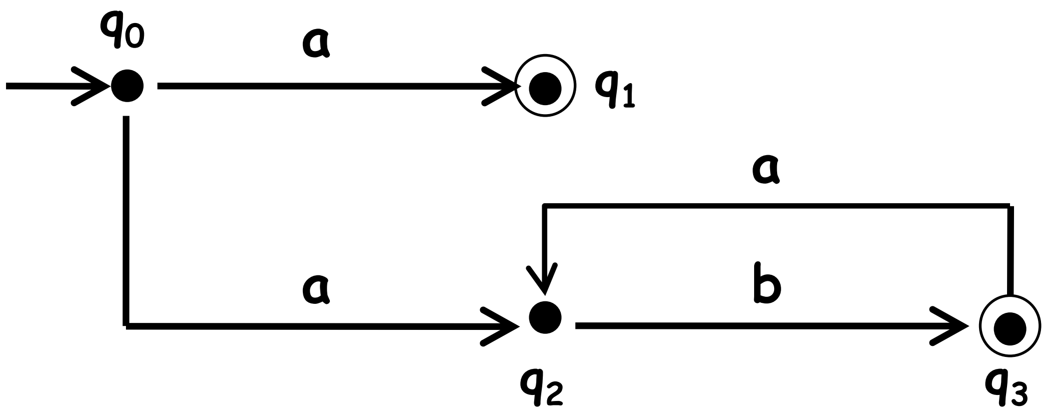
Todo AFN M' tiene un AFD M tal que $L(M')=L(M)$

Método para convertir un AFN en un AFD



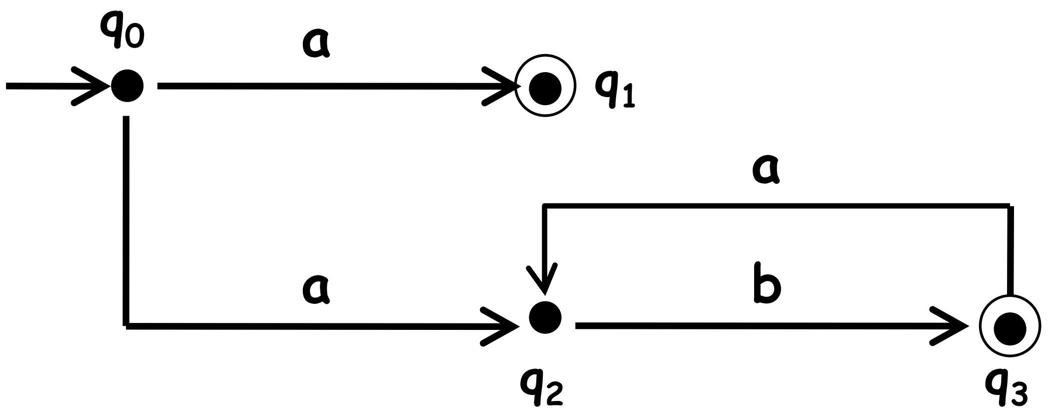
AFN que acepta $a \cup (ab)^+$

Método para convertir un AFN en un AFD



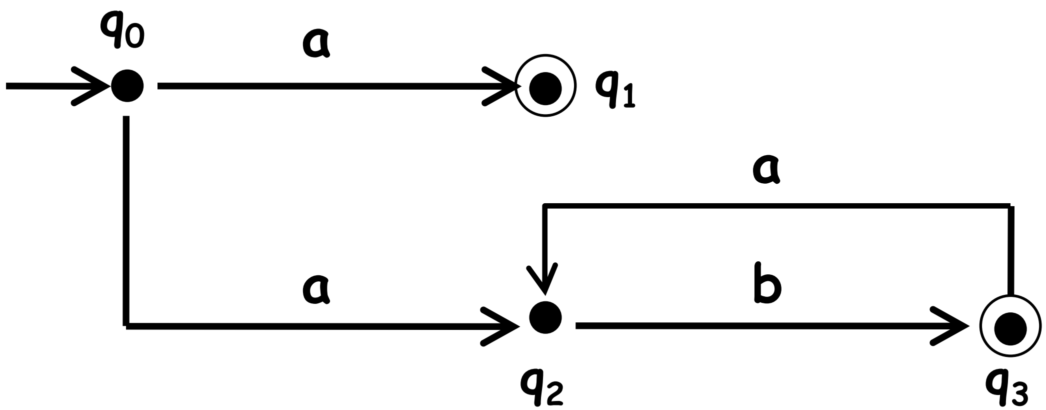
$\Delta(q_0,a)=\{q_1,q_2\}$
$\Delta(q_0,b)=\emptyset$

Método para convertir un AFN en un AFD



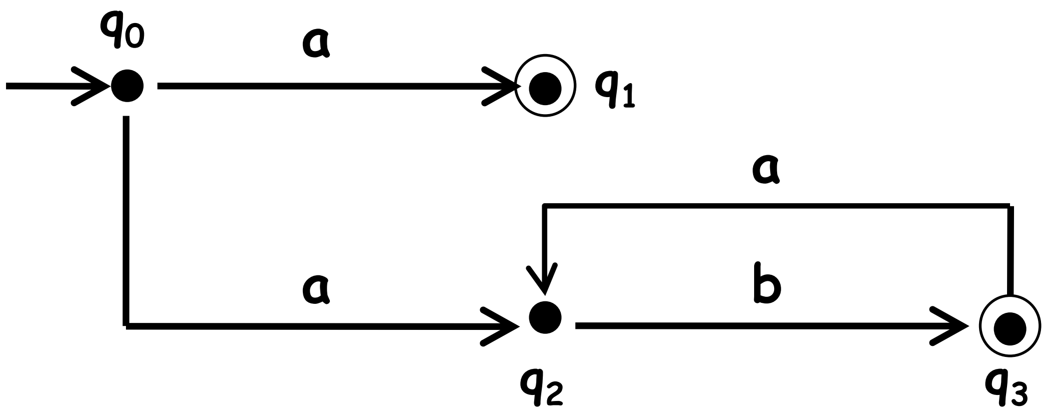
$\Delta(q_0,a)=\{q_1,q_2\}$
$\Delta(q_0,b)=\varnothing$
$\Delta(\{q_1,q_2\},a)=$
$\Delta(\{q_1,q_2\},b)=?$

Método para convertir un AFN en un AFD



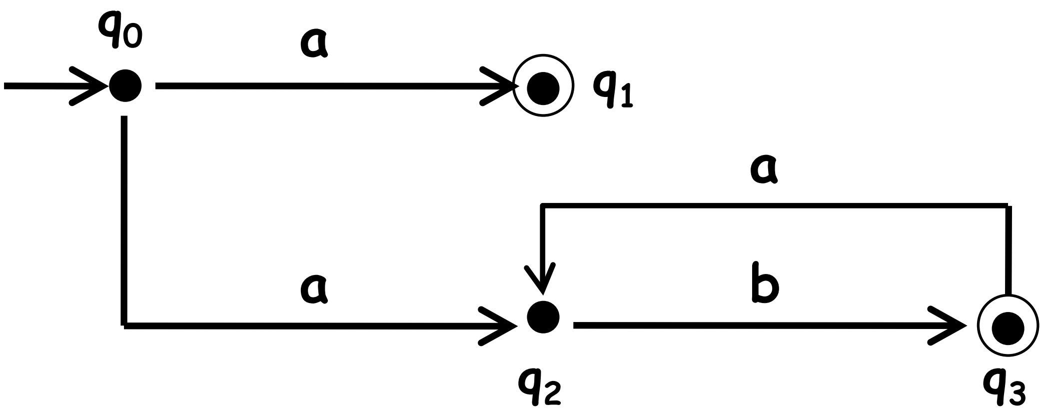
$\Delta(q_0,a)=\{q_1,q_2\}$
$\Delta(q_0,b)=\varnothing$
$\Delta(\{q_1,q_2\},a)=\varnothing$
$\Delta(\{q_1,q_2\},b)=\{q_3\}$

Método para convertir un AFN en un AFD



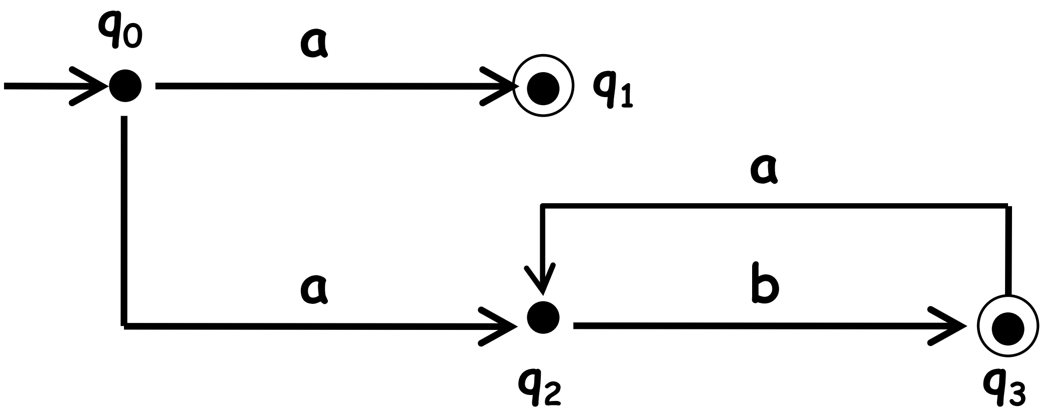
$\Delta(q_0,a)=\{q_1,q_2\}$
$\Delta(q_0,b)=\varnothing$
$\Delta(\{q_1,q_2\},a)=\varnothing$
$\Delta(\{q_1,q_2\},b)=\{q_3\}$
$\Delta(\{q_3\},a)=?$
$\Delta(\{q_3\},b)=?$

Método para convertir un AFN en un AFD



$\Delta(q_0,a)=\{q_1,q_2\}$
$\Delta(q_0,b)=\varnothing$
$\Delta(\{q_1,q_2\},a)=\varnothing$
$\Delta(\{q_1,q_2\},b)=\{q_3\}$
$\Delta(\{q_3\},a)=\{q_2\}$
$\Delta(\{q_3\},b)=\varnothing$

Método para convertir un AFN en un AFD



$\Delta(q_0,a)=\{q_1,q_2\}$

$\Delta(q_0,b)=\varnothing$

$\Delta(\{q_1,q_2\},a)=\varnothing$

$\Delta(\{q_1,q_2\},b)=\{q_3\}$

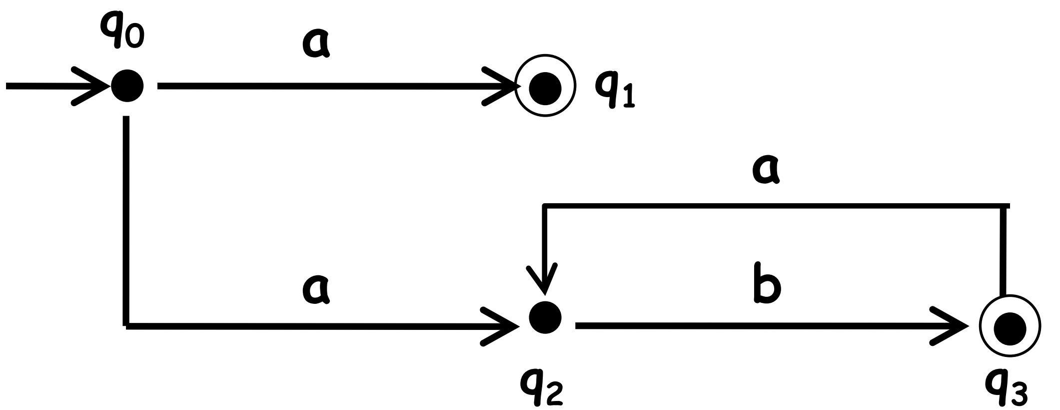
$\Delta(\{q_3\},a)=\{q_2\}$

$\Delta(\{q_3\},b)=\varnothing$

$\Delta(\{q_2\},a)=?$

$\Delta(\{q_2\},b)=?$

Método para convertir un AFN en un AFD



$\Delta(q_0,a)=\{q_1,q_2\}$
$\Delta(q_0,b)=\varnothing$
$\Delta(\{q_1,q_2\},a)=\varnothing$
$\Delta(\{q_1,q_2\},b)=\{q_3\}$
$\Delta(\{q_3\},a)=\{q_2\}$
$\Delta(\{q_3\},b)=\varnothing$

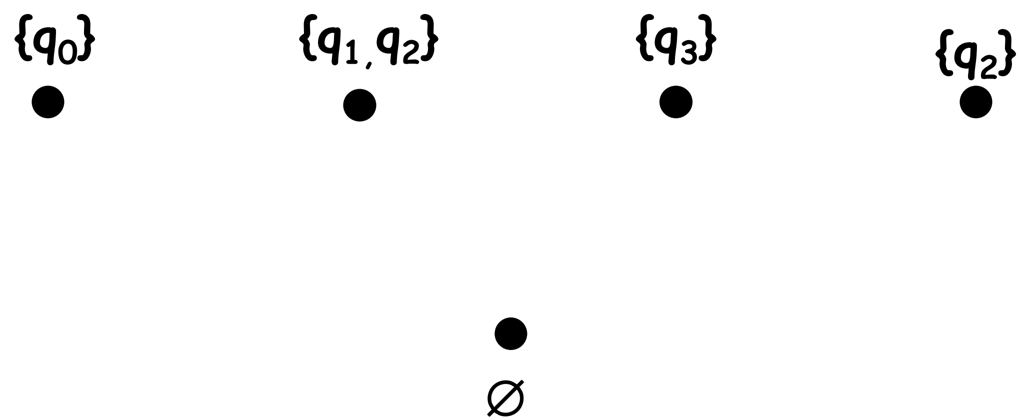
$\Delta(\{q_2\},a)=\varnothing$
$\Delta(\{q_2\},b)=\{q_3\}$

Método para convertir un AFN en un AFD

$\Delta(q_0,a)=\{q_1,q_2\}$
$\Delta(q_0,b)=\emptyset$
$\Delta(\{q_1,q_2\},a)=\emptyset$
$\Delta(\{q_1,q_2\},b)=\{q_3\}$
$\Delta(\{q_3\},a)=\{q_2\}$
$\Delta(\{q_3\},b)=\emptyset$
$\Delta(\{q_2\},a)=\emptyset$
$\Delta(\{q_2\},b)=\{q_3\}$

Método para convertir un AFN en un AFD

$\Delta(q_0,a)=\{q_1,q_2\}$
$\Delta(q_0,b)=\varnothing$
$\Delta(\{q_1,q_2\},a)=\varnothing$
$\Delta(\{q_1,q_2\},b)=\{q_3\}$
$\Delta(\{q_3\},a)=\{q_2\}$
$\Delta(\{q_3\},b)=\varnothing$
$\Delta(\{q_2\},a)=\varnothing$
$\Delta(\{q_2\},b)=\{q_3\}$



Método para convertir un AFN en un AFD

$\Delta(q_0,a)=\{q_1,q_2\}$

$\Delta(q_0,b)=\varnothing$

$\Delta(\{q_1,q_2\},a)=\varnothing$

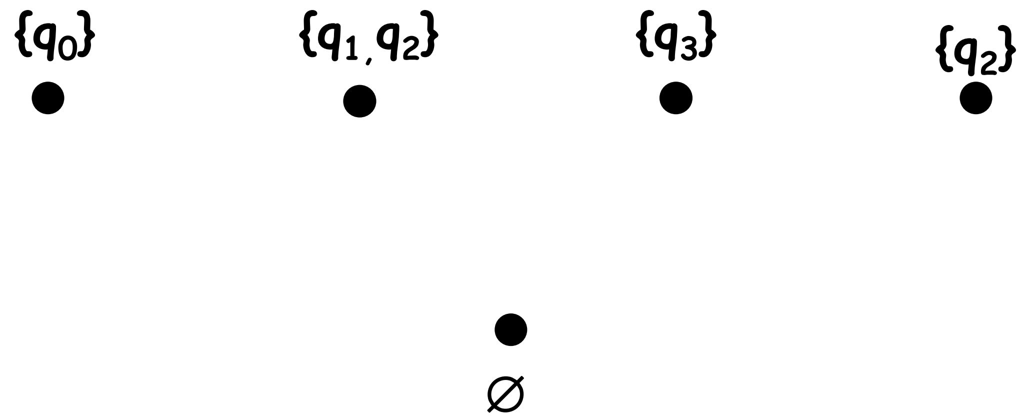
$\Delta(\{q_1,q_2\},b)=\{q_3\}$

$\Delta(\{q_3\},a)=\{q_2\}$

$\Delta(\{q_3\},b)=\varnothing$

$\Delta(\{q_2\},a)=\varnothing$

$\Delta(\{q_2\},b)=\{q_3\}$



- **Cualquier conjunto que contenga un estado de aceptación se marca como de aceptación**

Método para convertir un AFN en un AFD

$\Delta(q_0,a)=\{q_1,q_2\}$

$\Delta(q_0,b)=\varnothing$

$\Delta(\{q_1,q_2\},a)=\varnothing$

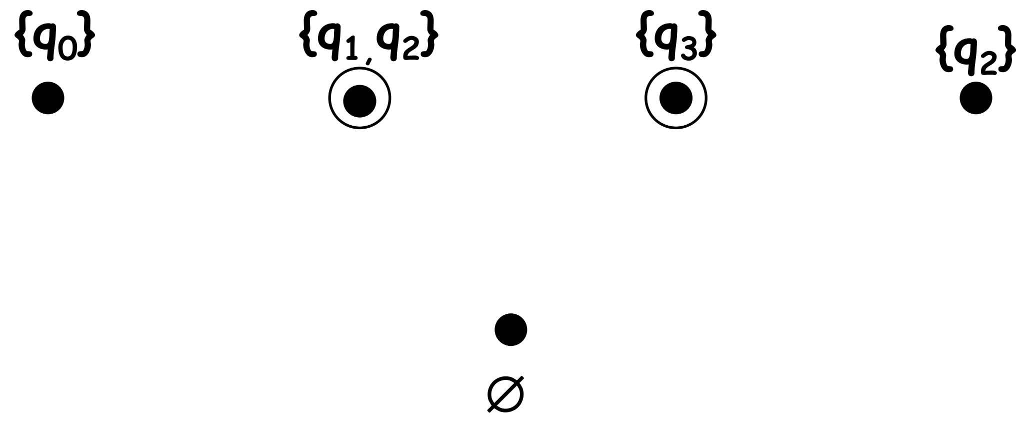
$\Delta(\{q_1,q_2\},b)=\{q_3\}$

$\Delta(\{q_3\},a)=\{q_2\}$

$\Delta(\{q_3\},b)=\varnothing$

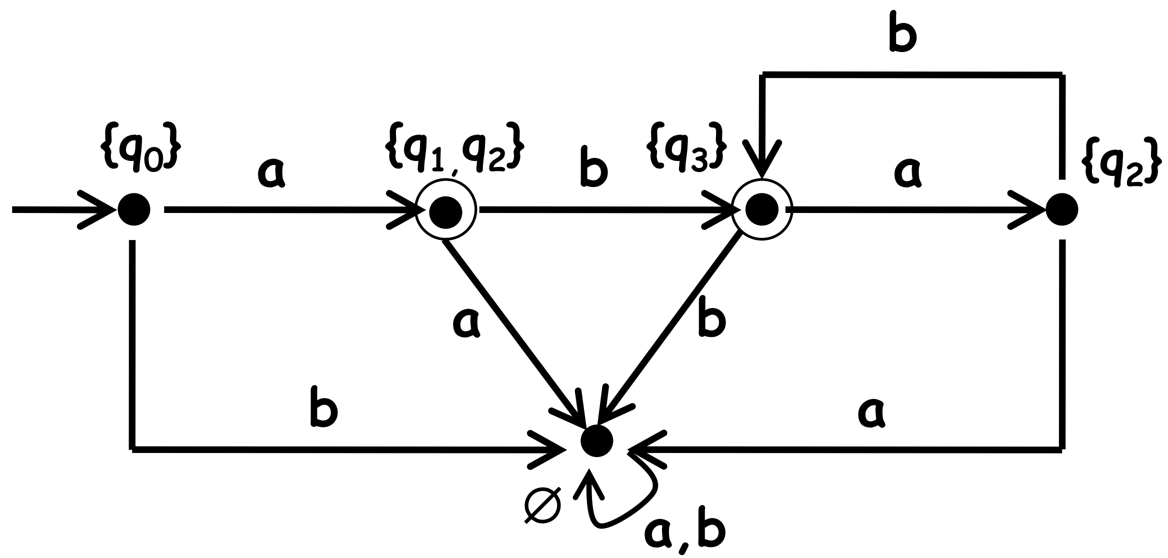
$\Delta(\{q_2\},a)=\varnothing$

$\Delta(\{q_2\},b)=\{q_3\}$



Método para convertir un AFN en un AFD

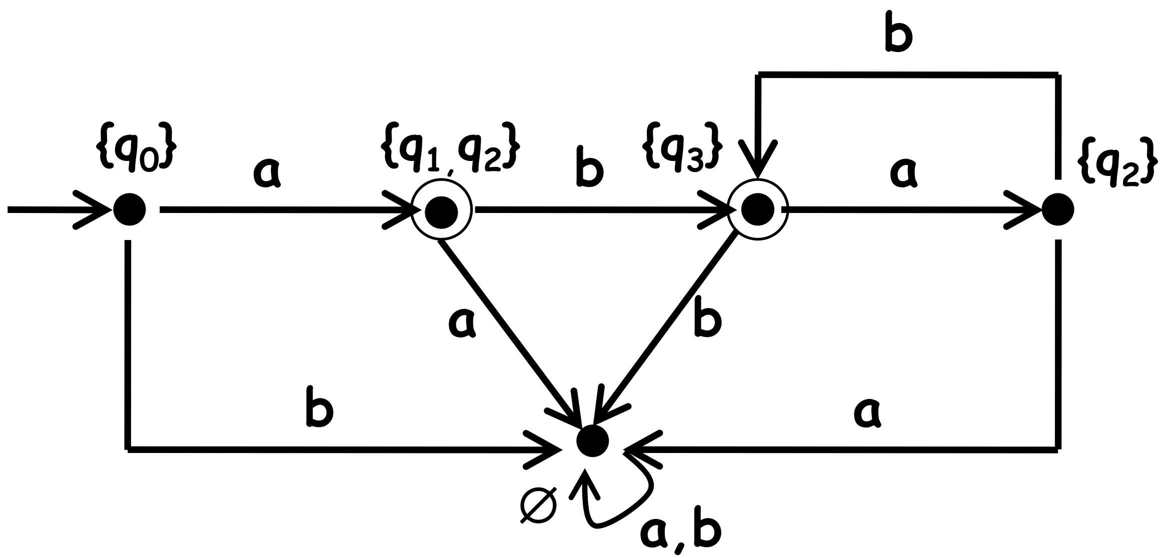
$\Delta(q_0,a)=\{q_1,q_2\}$
$\Delta(q_0,b)=\varnothing$
$\Delta(\{q_1,q_2\},a)=\varnothing$
$\Delta(\{q_1,q_2\},b)=\{q_3\}$
$\Delta(\{q_3\},a)=\{q_2\}$
$\Delta(\{q_3\},b)=\varnothing$
$\Delta(\{q_2\},a)=\varnothing$
$\Delta(\{q_2\},b)=\{q_3\}$



- **El nodo con etiqueta ∅ tiene transiciones que llegan a ese mismo nodo**

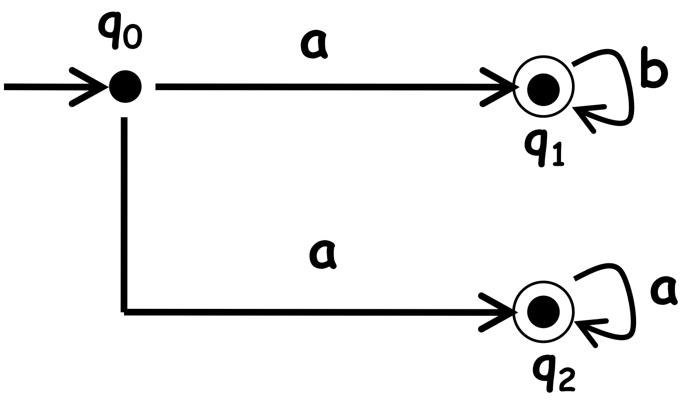
Método para convertir un AFN en un AFD

$\Delta(q_0,a)=\{q_1,q_2\}$
$\Delta(q_0,b)=\varnothing$
$\Delta(\{q_1,q_2\},a)=\varnothing$
$\Delta(\{q_1,q_2\},b)=\{q_3\}$
$\Delta(\{q_3\},a)=\{q_2\}$
$\Delta(\{q_3\},b)=\varnothing$
$\Delta(\{q_2\},a)=\varnothing$
$\Delta(\{q_2\},b)=\{q_3\}$



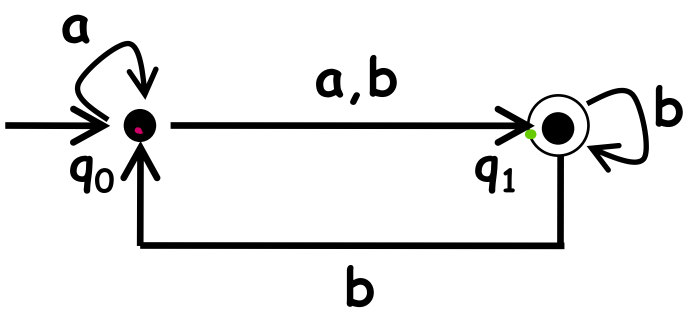
AFD que acepta $\mathbf{a}\cup(\mathbf{ab})^{+}$

Convierta el siguiente AFN a un AFD



AFN que acepta **ab***∪**a**$^+$

Convierta el siguiente AFN a un AFD



Lenguajes regulares

Convierta el siguiente AFN a un AFD