Sintaxis y Semántica del lenguaje.

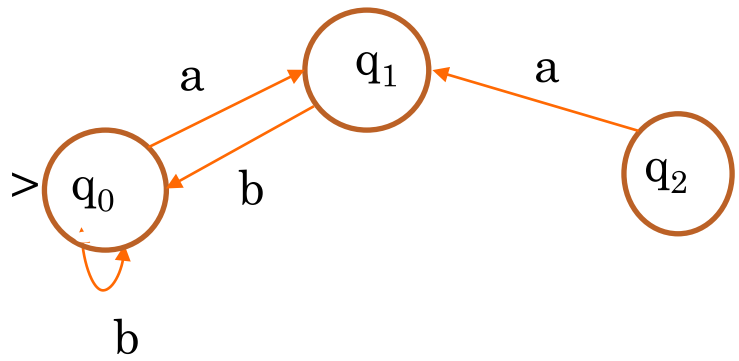
Autómatas Finitos:

Equivalencia y Minimización

Autómatas finitos. Equivalencia

Equivalencia de Autómatas Finitos:

<u>Estados accesibles</u>: En un AFD se dice que un estado p es accesible desde otro estado q, si existe un camino desde q hasta p, es decir, si llega alguna flecha.

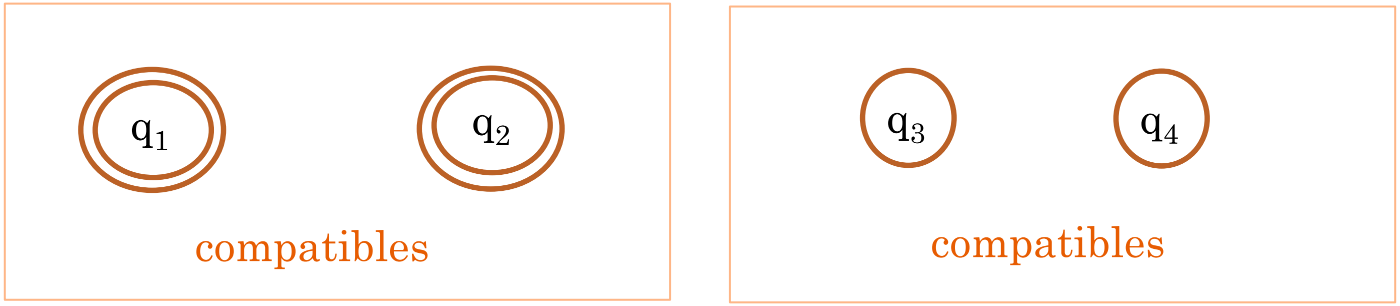


q2 es inaccesible

Todos los estados inaccesibles pueden ser eliminados sin que afecten el comportamiento del AF.

Autómatas finitos. Equivalencia

Estados compatibles: Dos estados q y q' son compatibles si ambos son estados finales o ambos son no finales.



Autómatas finitos equivalentes: Dos autómatas finitos M y M' son equivalentes si aceptan las mismas palabras, es decir el mismo lenguaje.

Autómatas finitos. Equivalencia

Teorema de Moore: existe un algoritmo de comparación para decidir si 2 AF son equivalentes o no.

Para comparar los AF se construye un árbol de comparación, siguiendo los siguientes pasos:

0- Se eliminan estados inaccesibles.

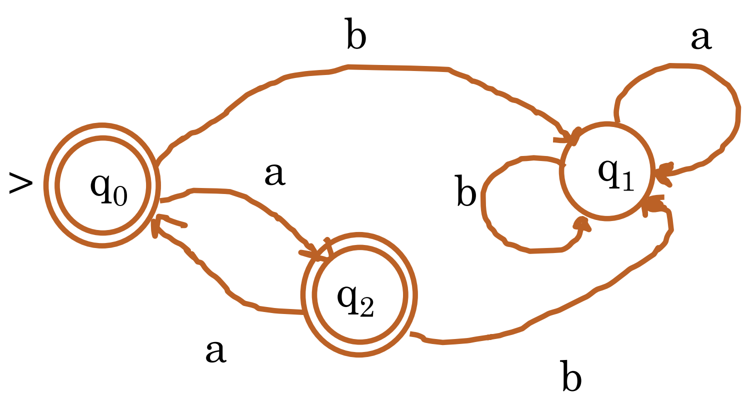
1- La raíz del árbol es el par ordenado (S,S') formado por los estados iniciales de cada AF. Los estados deben ser compatibles.

2- Luego se van agregando hojas del árbol usando las transiciones que salen desde S y S'. Cada par ordenado de estados que se agrega debe ser compatible.

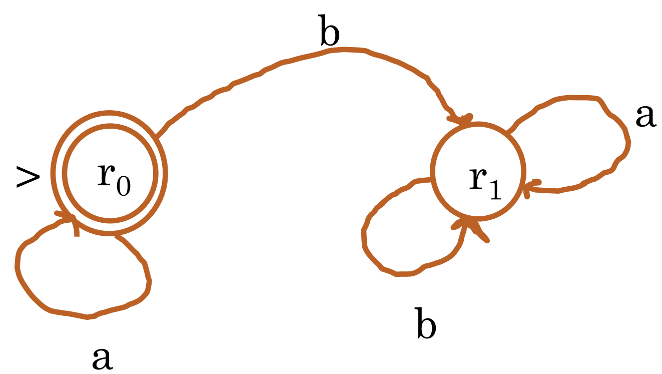
Autómatas finitos. Equivalencia

Ejemplo1:

AF1:



AF2:

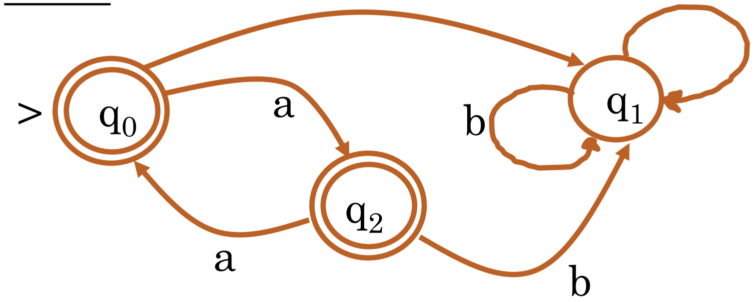


Dados dos autómatas AF1 y AF2,

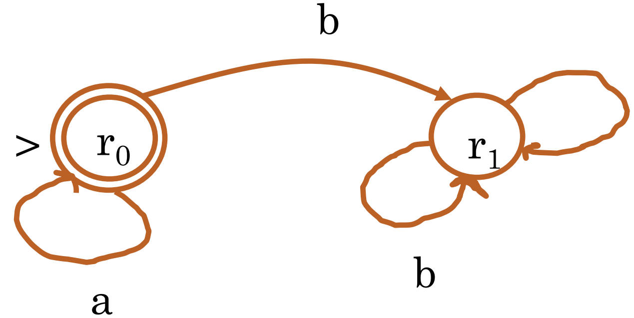
¿cómo determinamos si son equivalentes?

AUTÓMATAS FINITOS. EQUIVALENCIA

AF1:



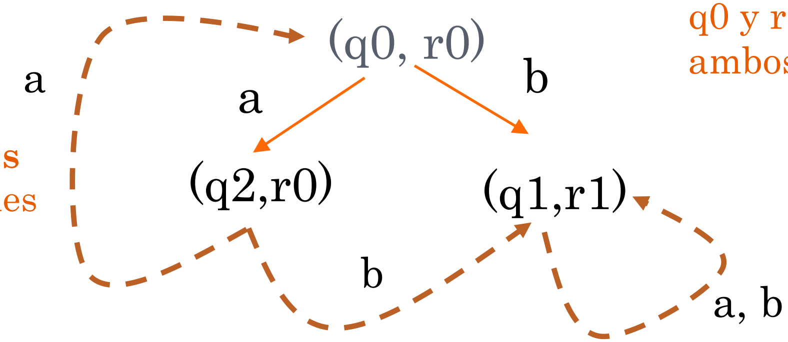
AF2:



Aplicamos el teorema de Moore, siguiendo los pasos indicados anteriormente.

No hay estados inaccesibles, entonces construyo la raíz del árbol con los estados iniciales de ambos autómatas (q0,r0)

q0 y r0 son **compatibles** porque ambos son finales, seguimos



q_2 y r_0 son **compatibles** porque ambos son finales entonces seguimos

q_1 y r_1 son **compatibles** porque ambos son no finales entonces seguimos

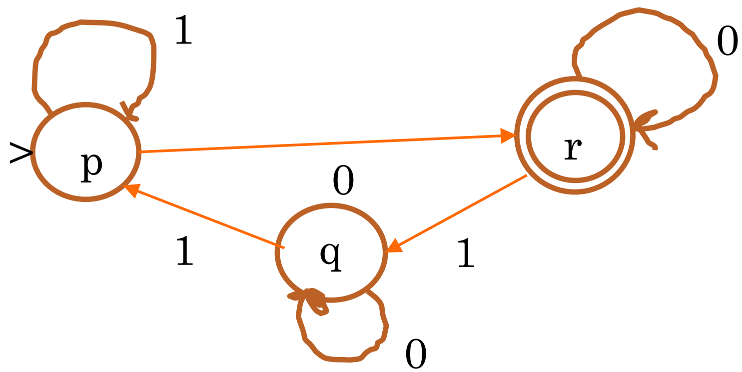
Todos los estados son compatibles. Por lo tanto, AF1, AF2 **son equivalentes.**

Con línea punteada se marcan las ramas que van a nodos ya existentes.

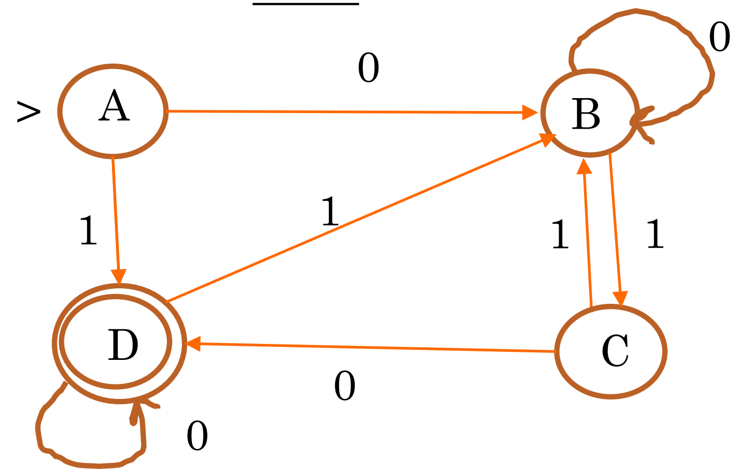
Autómatas finitos. Equivalencia

<u>Ejemplo 2:</u>

<u>AF1:</u>



<u>AF2:</u>



No hay estados inaccesibles, entonces armo el árbol de comparación con los estados iniciales

P y A son **compatibles** porque ambos son no finales, seguimos

(p, A)

$\swarrow$ 0

(r, B)

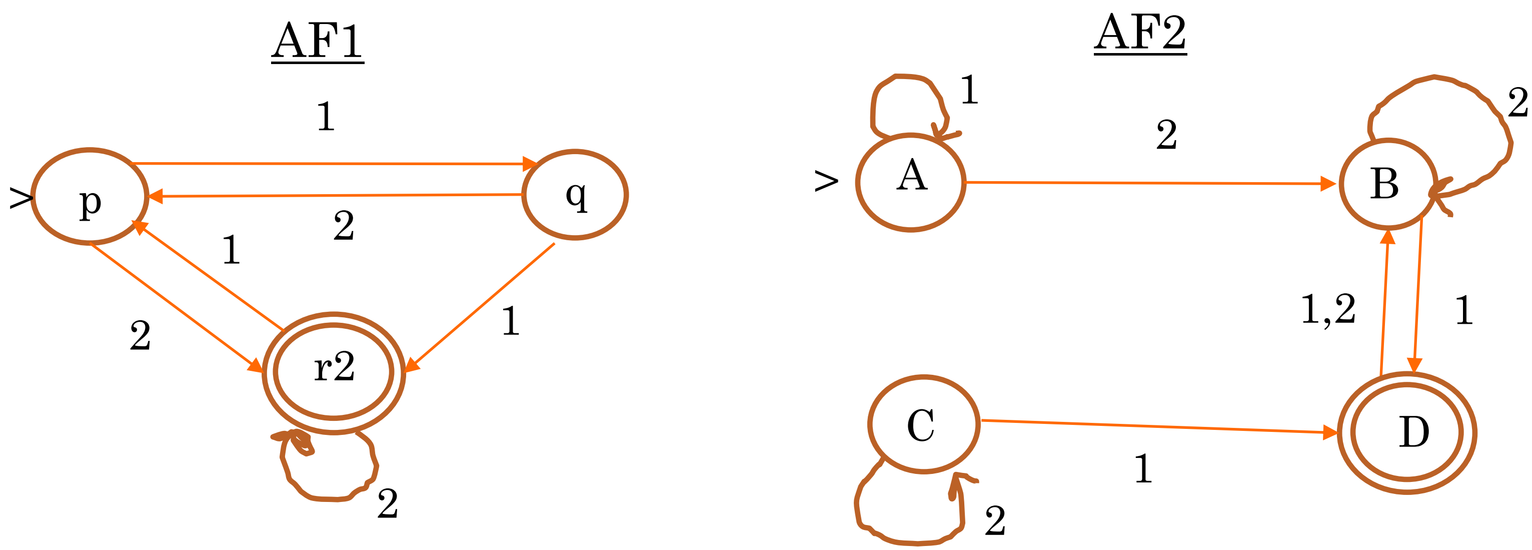
Como r y B no son estados **compatibles**, porque r es estado final y B no final, no seguimos.

Por lo tanto, AF1, AF2 **no** son equivalentes.

Si encuentro un par de estados incompatibles, entonces ya no son equivalentes.
No es necesario completar el árbol si ya se encontró un par incompatible.

Autómatas finitos. Equivalencia

Ejemplo 3:



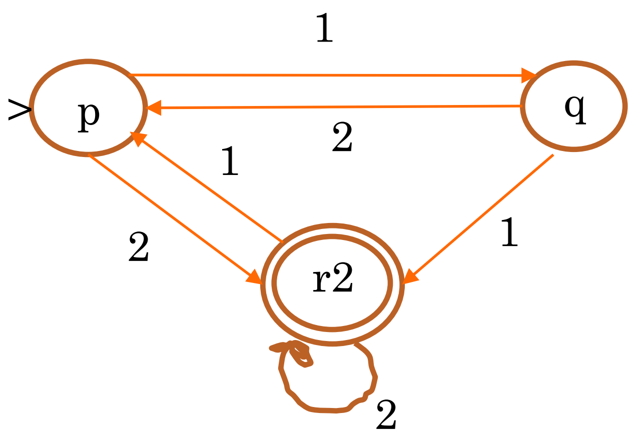
1° saco los nodos inaccesibles de AF2.

C es inaccesible, porque no llega ninguna flecha a él. Se elimina.

Autómatas finitos. Equivalencia

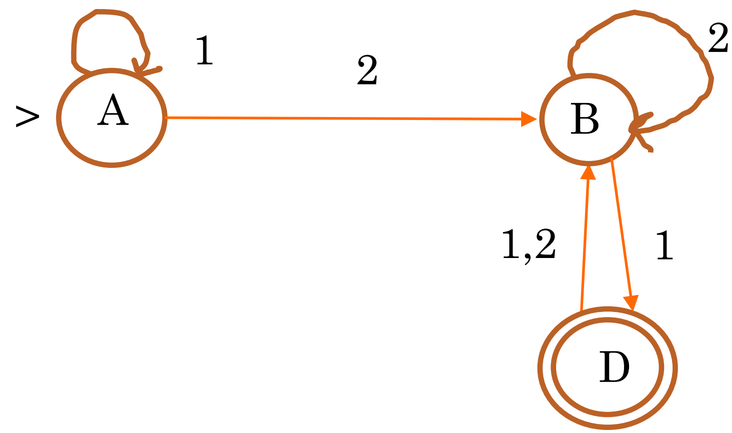
<u>Ejemplo 3:</u>

<u>AF1</u>



Luego de la eliminación de nodos inaccesibles AF2 queda así:

<u>AF2</u>



Armo el árbol de comparación, a partir de los estados iniciales.

(p, A)

1 (q,A)

2 (r2,B)

P y A son **compatibles** porque son ambos no finales, seguimos

q y A son **compatibles** porque son ambos no finales, seguimos

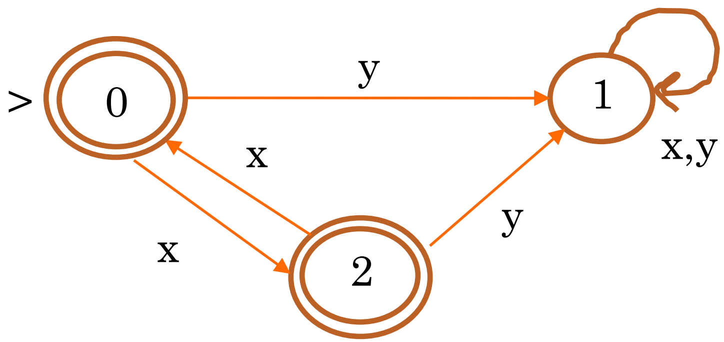
r2y B son **incompatibles** porque r2 es final y B es no final, entonces no se continúa

Por lo tanto, AF1, AF2 **no** son equivalentes.

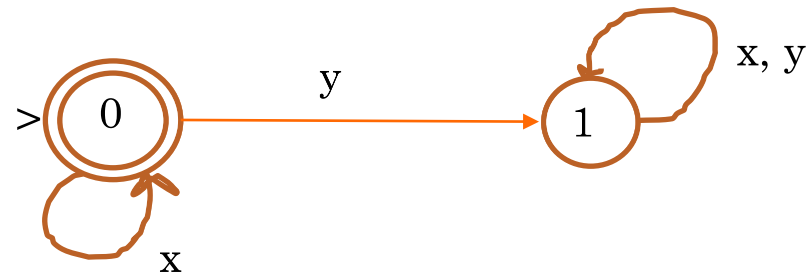
Autómatas finitos. Equivalencia

Ejemplo 4:

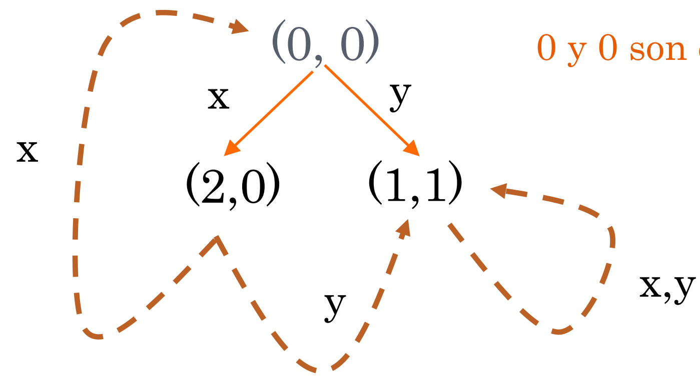
AF1



AF2



No hay estados inaccesibles, entonces armo el árbol de comparación con los estados inicales



0 y 0 son **compatibles**

2 y 0 son **compatibles**

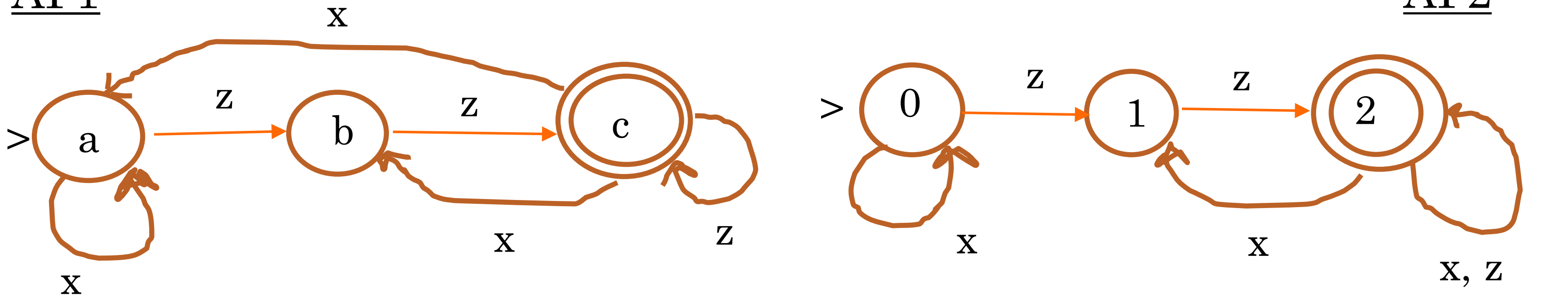
1 y 1 son **compatibles**

Todos los estados son compatibles. Por lo tanto, AF1, AF2 **son equivalentes.**

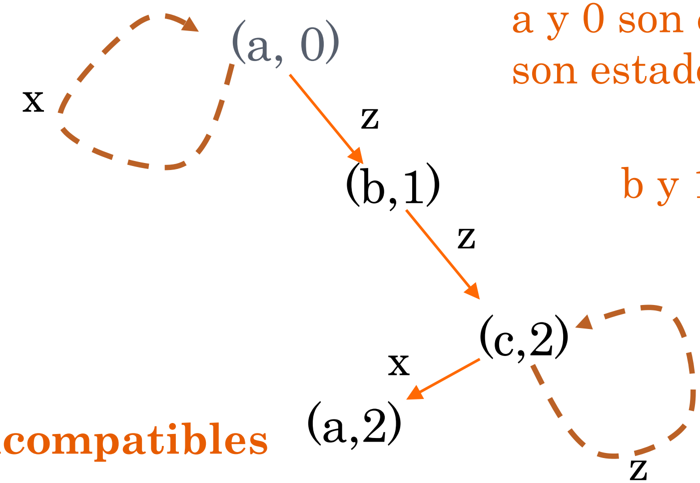
Autómatas finitos. Equivalencia

Ejemplo 5:

AF1 AF2



No hay estados inaccesibles, entonces armo el árbol de comparación con los estados iniciales



a y 0 son **compatibles** porque ambos son estados no finales, seguimos

b y 1 son **compatibles**

c y 2 son **compatibles**

a y 2 son **incompatibles**

Por lo tanto, AF1, AF2 **no son equivalentes.**

AUTÓMATAS FINITOS. MINIMIZACION

Minimización de AFD:

Existen métodos mecánicos para simplificar un AFD para llegar a la expresión más simple del mismo.

Se quitan estados inaccesibles y estados indistinguibles o redundantes.

El autómata mínimo cumple su función con la menor cantidad posible de estados.

Dos **estados** son **distinguibles** si uno es final y el otro no.

q_1 y q_2 son distinguibles

Autómatas finitos. minimizacion

Método para Minimizar un AFD:

El método consiste en eliminar estados inaccesibles (inútiles) y los redundantes o equivalentes.

Para detectar los estados redundantes hay que ver si son distinguibles entre sí y sustituirlos por un único estado.

Proceso:

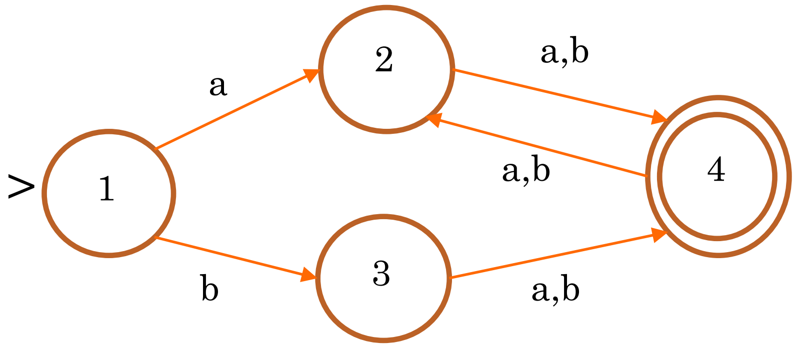
1) se buscan estados inaccesibles y se eliminan.
2) se marcan los estados distinguibles entre sí, finales con no finales.
3) se analiza cada par de estados restante,

   si dos estados p y q de llegada son distinguibles entonces los estados de partida p_0 y q_0 también lo son.

Para realizar el proceso se arma una tabla de estados, con cada estado como fila y columna.

Autómatas finitos. minimizacion

<u>Ejemplo 1:</u>



	1	2	3	4
1	X	X	X	X
2		X	X	X
3			X	X
4	//	//	//	X

En este ejemplo no hay estados inaccesibles.

Se procede a armar la tabla, mediante los siguientes pasos:

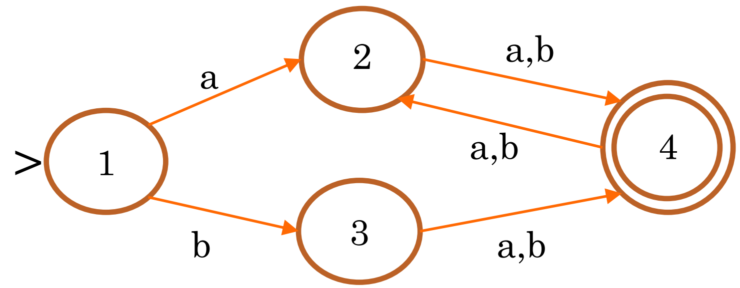
1) se descarta la mitad superior de la tabla y se marcan las celdas con una X.

2) A partir del grafo se marcan los pares distinguibles, como 4 es estado final y 1, 2 y 3 son no finales, entonces los pares distinguibles son:

(1,4), (2,4), (3,4), se marcan en la tabla con //

Autómatas finitos. minimizacion

Ejemplo 1:



	1	2	3	4
1	X	X	X	X
2		X	X	X
3	///		X	X
4	//	//	//	X

3) A partir de la tabla, se la recorre de abajo hacia arriba tomando las casillas no tachadas: (3,1), (3,2) y (2,1)

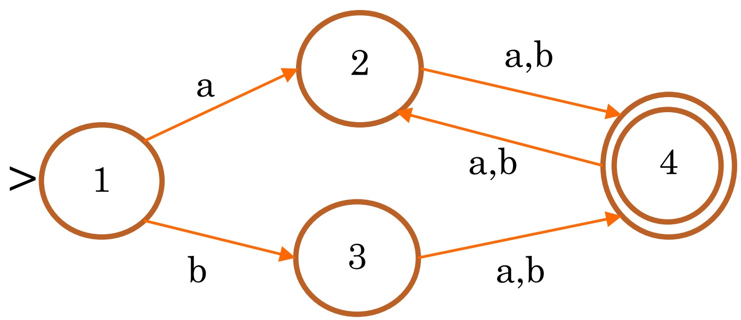
De cada par, se evalúan sus elementos con cada transición:

(3,1) ? 3,a → 4
1,a → 2
distinguible
(3,1) es distinguible

(3,2) ? 3,a → 4
2,a → 4
?
3,b → 4
2,b → 4
?
No se marca en la tabla!!

Autómatas finitos. minimizacion

<u>Ejemplo 1:</u>



	1	2	3	4
1	X	X	X	X
2	////	X	X	X
3	///		X	X
4	//	//	//	X

(2,1) ? 2,a → 4
1,a → 2

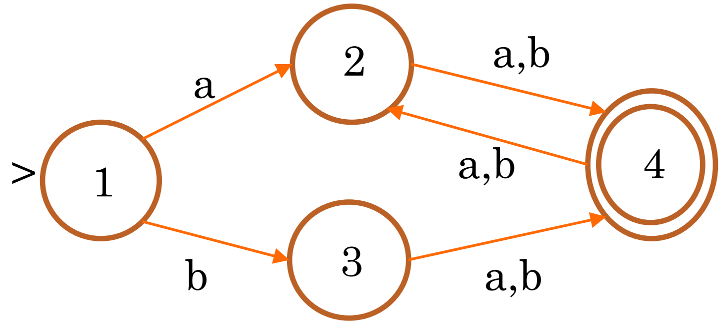
distinguible

(2,1) es distinguible

Ya no quedan mas pares a evaluar. Se construye entonces el AFD mínimo.

Autómatas finitos. minimizacion

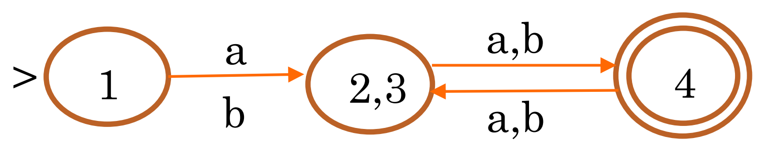
Ejemplo 1:



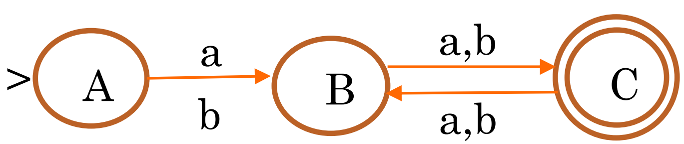
Se agrupan los estados indistinguibles: (3,2).

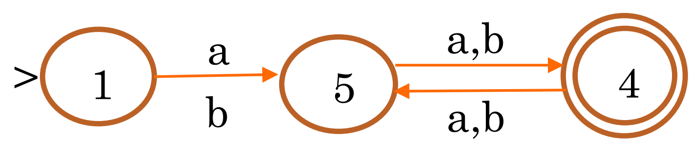
	1	2	3	4
1	X	X	X	X
2	////	X	X	X
3	///		X	X
4	//	//	//	X

Se reescribe el autómata considerando a (2,3) o (3,2) como un único estado y se mantienen las transiciones del AFD original.



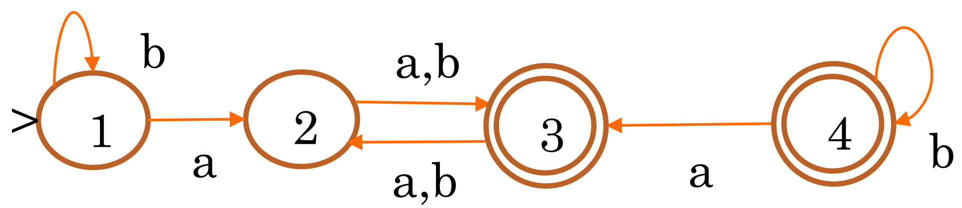
Algunos autores renombran el autómata de esta forma:





Autómatas finitos. minimizacion

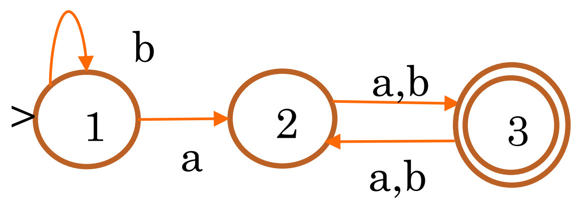
Ejemplo 2:



1°) Se eliminan los estados inaccesibles.

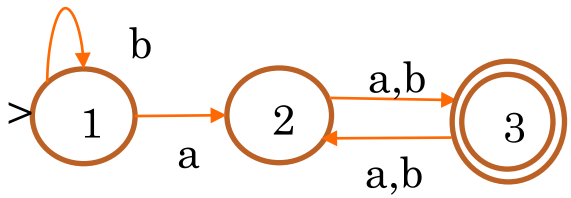
El estado 4 es inaccesible.

Se reescribe el autómata sin el estado 4.



Autómatas finitos. minimizacion

Ejemplo 2:



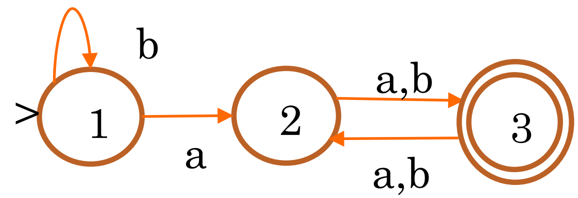
	1	2	3
1	X	X	X
2		X	X
3	//	//	X

Se procede a armar la tabla, luego se siguen los siguientes pasos:

1) se descarta la mitad superior de la tabla y se marcan las celdas con una X.
2) se marcan los pares distinguibles, como 3 es estado final y 1, 2 no finales, entonces los pares distinguibles son: (3,1), (3,2) se marcan en la tabla con //

AUTÓMATAS FINITOS. MINIMIZACION

<u>Ejemplo 2:</u>



	1	2	3
1	X	X	X
2	///	X	X
3	//	//	X

Vamos a ver que pasa con (2,1)

3) (2,1) ? 2,a $\rightarrow$ 3

1,a $\rightarrow$ 2

distinguible

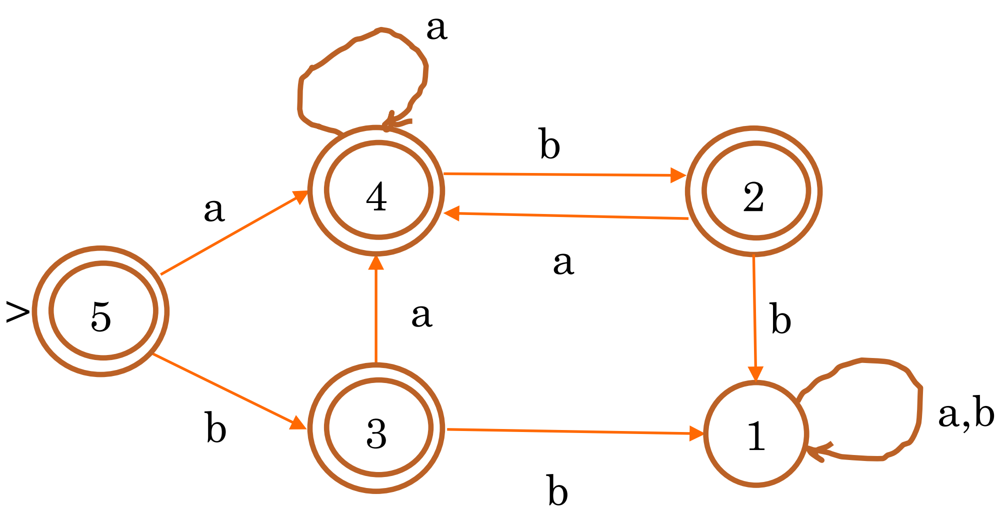
(2,1)
es distinguible

Por lo tanto al autómata finito es mínimo.

Autómatas finitos. minimizacion

Ejemplo 3:

	1	2	3	4	5
1	X	X	X	X	X
2	//	X	X	X	X
3	//		X	X	X
4	//			X	X
5	//				X



1°) No hay estados inaccesibles.

2°) Se procede a armar la tabla, luego se siguen los siguientes pasos:

1) se descarta la mitad superior de la tabla y se marcan las celdas con una X.

2) se marcan los pares distinguibles. Estados finales: 5, 4, 3, 2.

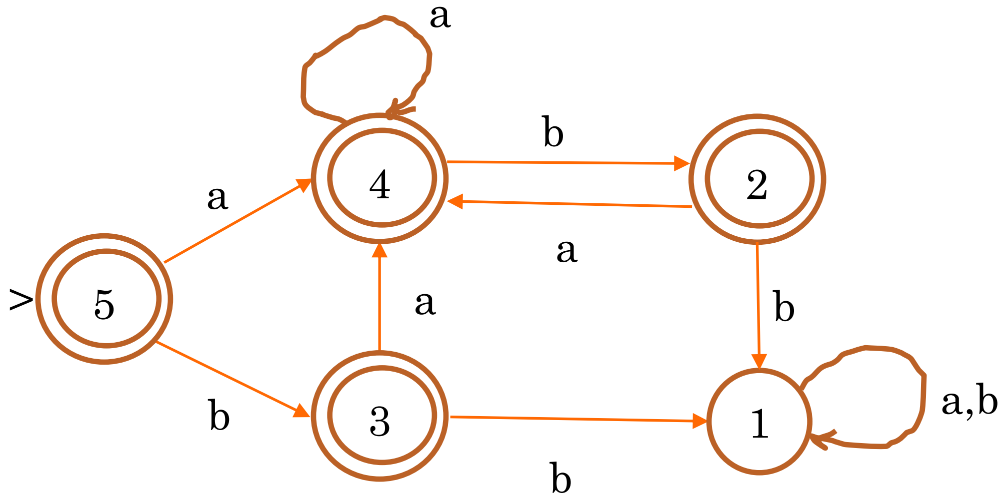
Estados no finales: 1

Pares distinguibles: (1,5), (1,4), (1,3), (1,2), se marcan en la tabla con //

AUTÓMATAS FINITOS. MINIMIZACION

Ejemplo 3:

	1	2	3	4	5
1	X	X	X	X	X
2	//	X	X	X	X
3	//		X	X	X
4	//			X	X
5	//	///			X



Analizo el resto de la tabla:

Vamos a ver que pasa con (5,2)

3) (5,2) ? 5,a → 4

2,a → 4 ?

(5,2) es distinguible y lo marco en la tabla

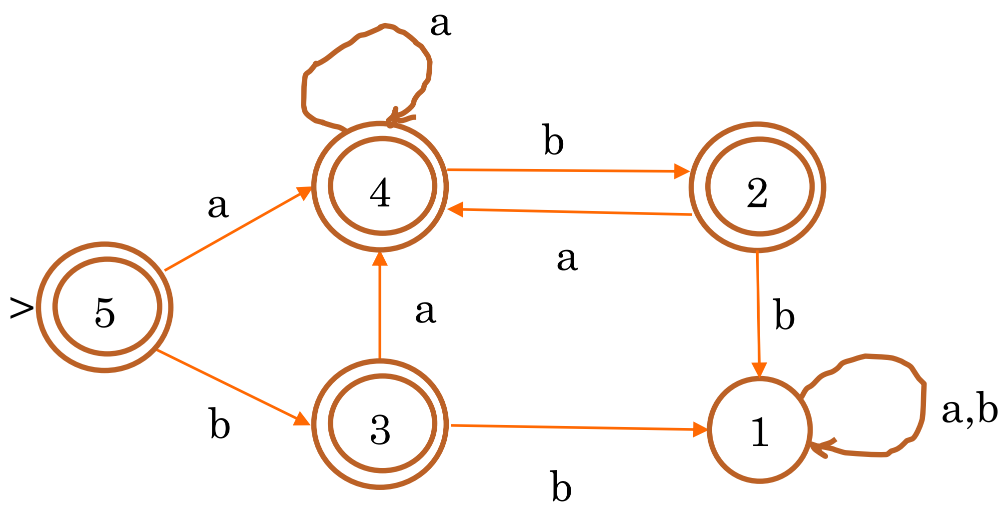
5,b 3

2,b 1 es distinguible

Autómatas finitos minimizacion

Ejemplo 3:

	1	2	3	4	5
1	X	X	X	X	X
2	//	X	X	X	X
3	//		X	X	X
4	//			X	X
5	//	///	///		X



Vamos a ver que pasa con (5,3)

3) (5,3) ?

5,a → 4

3,a → 4

?

5,b → 3

3,b → 1

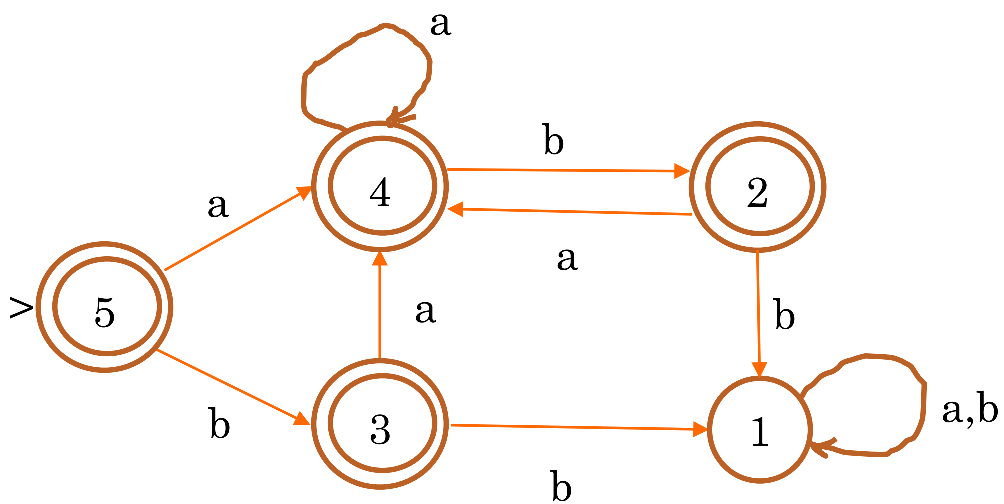
distinguible

(5,3) es distinguible y lo marco en la tabla

Autómatas finitos. minimizacion

Ejemplo 3:

	1	2	3	4	5
1	X	X	X	X	X
2	//	X	X	X	X
3	//		X	X	X
4	//			X	X
5	//	///	///		X



Vamos a ver que pasa con (5,4)

3) (5,4) ? 5,a → 4

4,a → 4 ?

5,b → 3

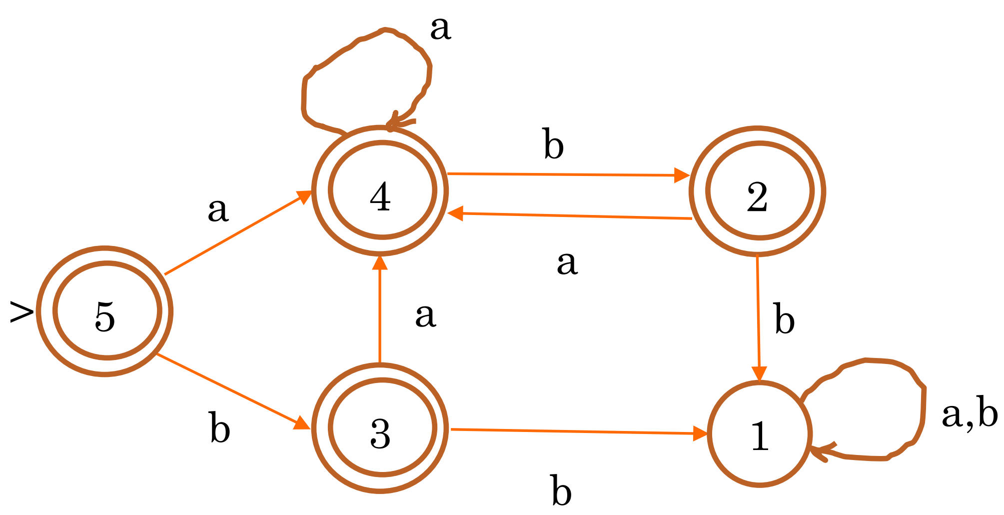
4,b → 2 ?

(5,4) es indistinguible, no lo marco en la tabla

Autómatas finitos. minimizacion

Ejemplo 3:

	1	2	3	4	5
1	X	X	X	X	X
2	//	X	X	X	X
3	//		X	X	X
4	//	////		X	X
5	//	///	///		X



Vamos a ver que pasa con (4,2)

4) (4,2) ? $4,a \rightarrow 4$

$2,a \rightarrow 4$?

$4,b \rightarrow 2$

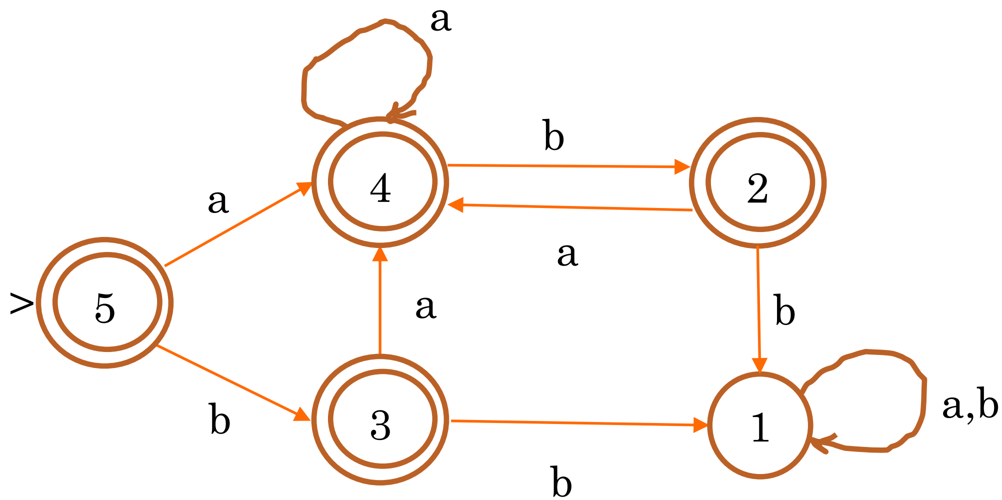
$2,b \rightarrow 1$ distinguible

(4,2)
es distinguible, lo marco en la tabla

Autómatas finitos. minimizacion

Ejemplo 3:

	1	2	3	4	5
1	X	X	X	X	X
2	//	X	X	X	X
3	//		X	X	X
4	//	////	////	X	X
5	//	///	///		X



Vamos a ver que pasa con (4,3)

4) (4,3) ? 4,a → 4 ; 3,a → 4 ?

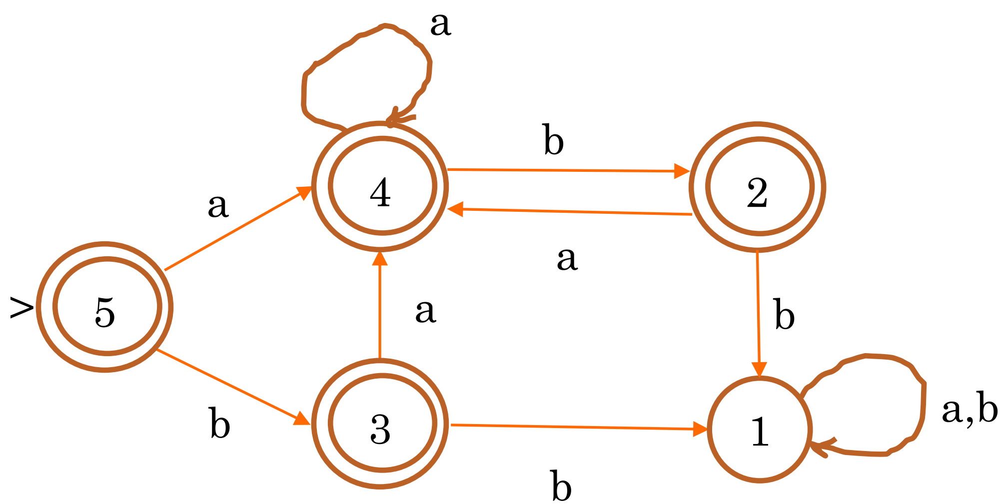
4,b → 2 ; 3,b → 1 distinguible

(4,3)
es distinguible, lo marco en la tabla

AUTÓMATAS FINITOS. MINIMIZACION

Ejemplo 3:

	1	2	3	4	5
1	X	X	X	X	X
2	//	X	X	X	X
3	//		X	X	X
4	//	////	////	X	X
5	//	///	///		X



Vamos a ver que pasa con (3,2)

4) (3,2) ? 3,a → 4

2,a → 4 ?

3,b → 1

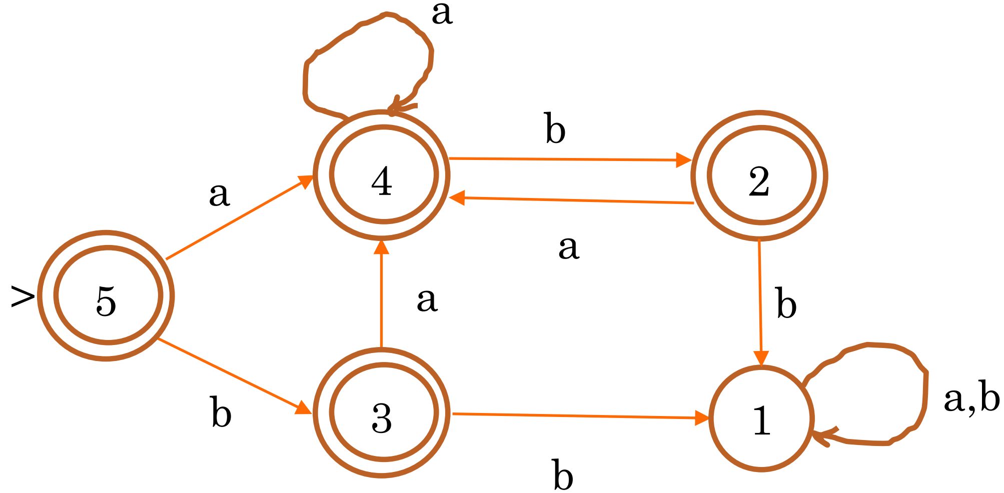
2,b → 1 ?

(3,2)
es indistinguible, **no** lo marco en la tabla

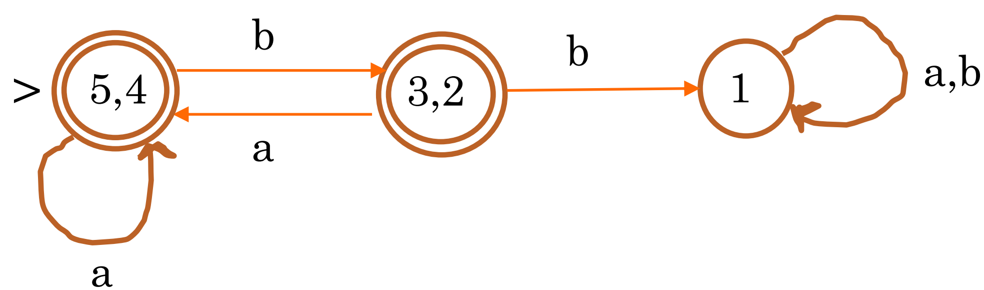
Autómatas finitos. minimizacion

Ejemplo 3:

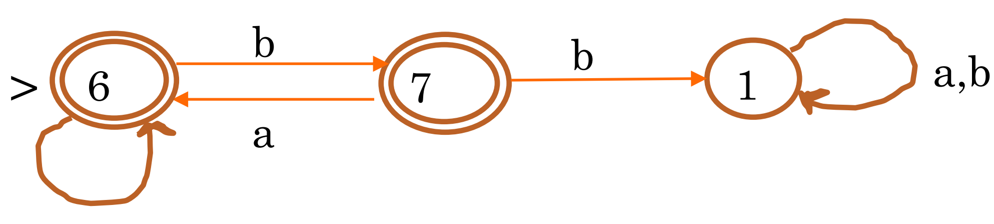
	1	2	3	4	5
1	X	X	X	X	X
2	//	X	X	X	X
3	//		X	X	X
4	//	////	////	X	X
5	//	///	///		X



Luego, se agrupan los estados indistinguibles. Se reescribe el autómata considerando tanto a (5,4) como a (3,2) como estados únicos.

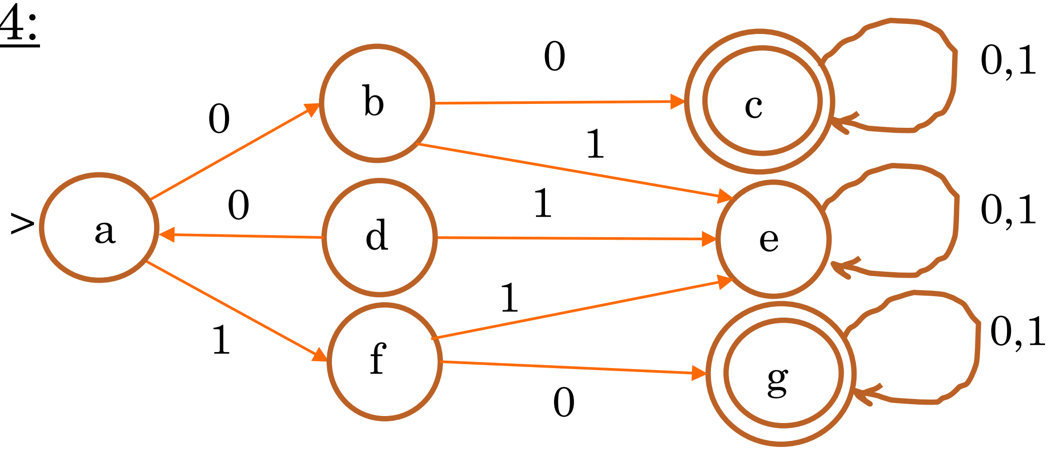


Por último se renombra el autómata de esta forma:



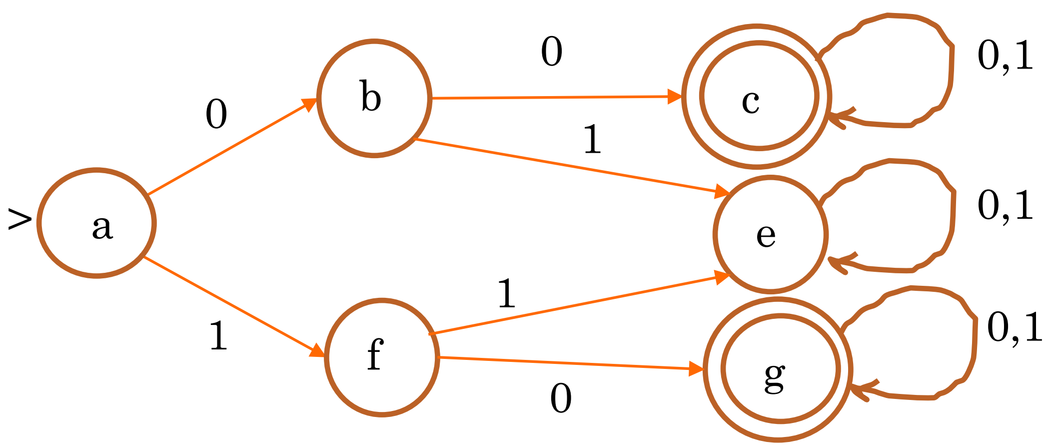
Autómatas finitos. minimizacion

Ejemplo 4:



1°) Elimino los estados inaccesibles y reescribo el autómata.

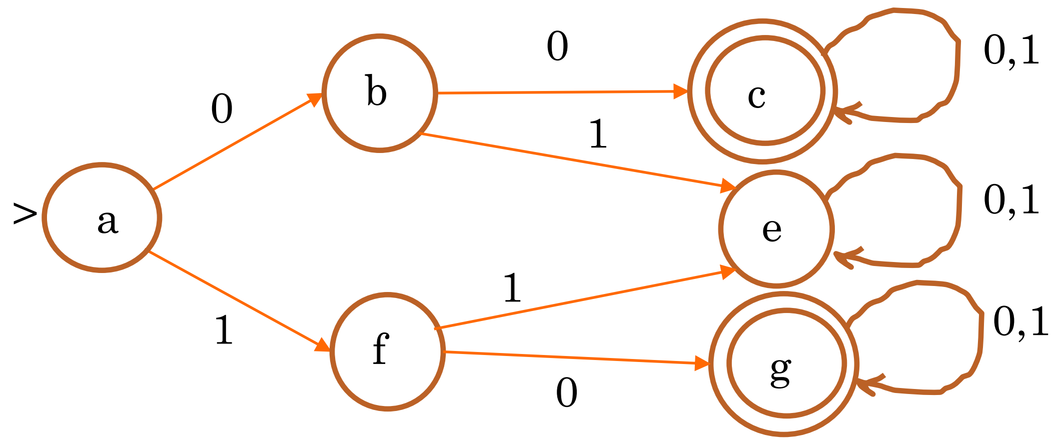
d es inaccesible, por lo tanto elimino ese estado del autómata.



AUTÓMATAS FINITOS. MINIMIZACION

<u>Ejemplo 4:</u>

	a	b	c	e	f	g
a	X	X	X	X	X	X
b		X	X	X	X	X
c	//	//	X	X	X	X
e			//	X	X	X
f			//		X	X
g	//	//		//	//	X



No aparece d en la tabla porque se eliminó.

2°) Se arma la tabla, luego se siguen los siguientes pasos:

1) se descarta la mitad superior de la tabla y se marcan las celdas con una X.

2) se marcan los pares distinguibles, Estados finales: c y g.

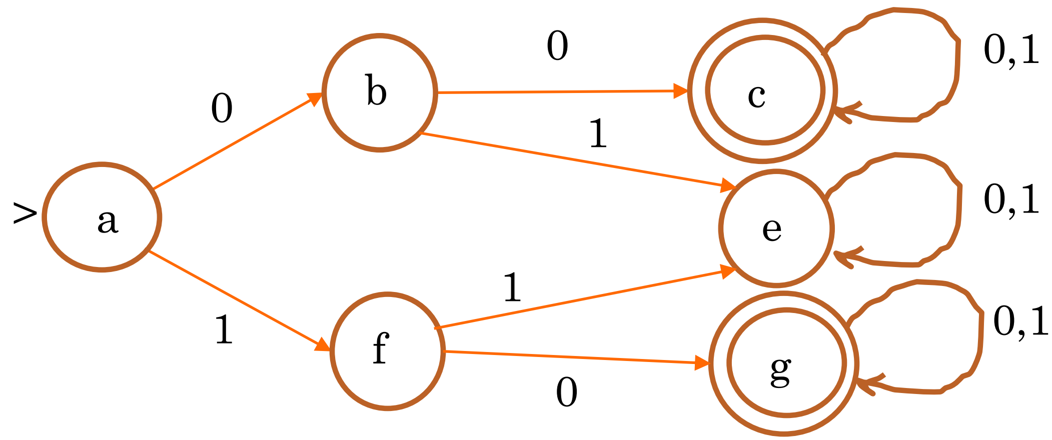
Estados no finales: a,b,e,f

Pares distinguibles: (a,c), (b,c), (e,c), (f,c), (a,g), (b,g), (e,g), (f,g), se marcan en la tabla con //

Autómatas finitos. minimizacion

Ejemplo 4:

	a	b	c	e	f	g
a	X	X	X	X	X	X
b		X	X	X	X	X
c	//	//	X	X	X	X
e			//	X	X	X
f			//		X	X
g	//	//		//	//	X



Analizo el resto de la tabla:

Vamos a ver que pasa con (g,c)

3) (g,c) ?

g,0 → g

c,0 → c

?

g,1 → g

c,1 → c

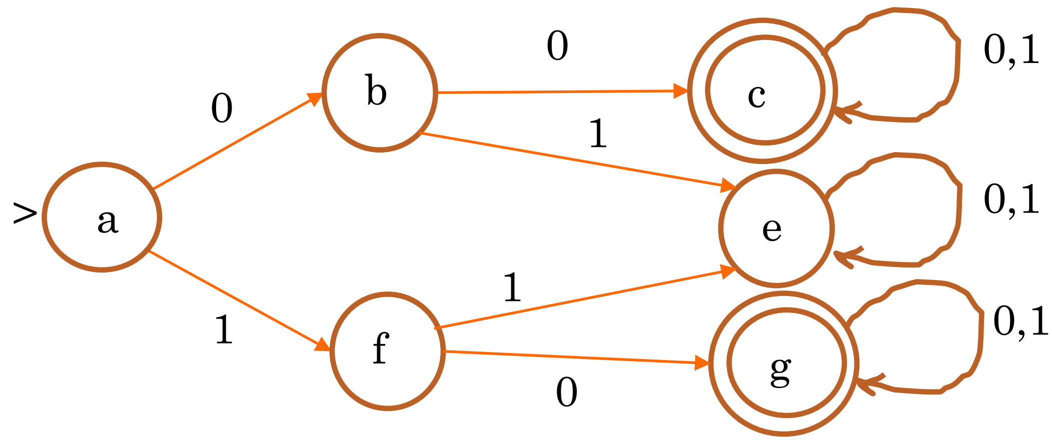
?

(g,c)
es indistinguible **no** lo marco en la tabla

Autómatas finitos. minimizacion

Ejemplo 4:

	a	b	c	e	f	g
a	X	X	X	X	X	X
b		X	X	X	X	X
c	//	//	X	X	X	X
e			//	X	X	X
f	///		//		X	X
g	//	//		//	//	X



Analizo el resto de la tabla:

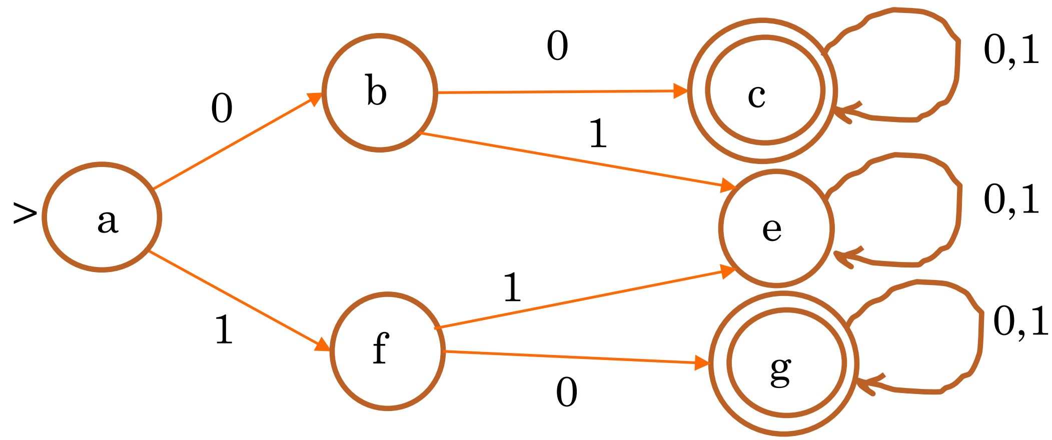
Vamos a ver que pasa con (f,a)

4) (f,a) ? f,0 → g ; a,0 → b — distinguible — (f,a) es distinguible, lo marco en la tabla

Autómatas finitos. minimizacion

<u>Ejemplo 4:</u>

	a	b	c	e	f	g
a	X	X	X	X	X	X
b		X	X	X	X	X
c	//	//	X	X	X	X
e			//	X	X	X
f	///		//		X	X
g	//	//		//	//	X



Analizo el resto de la tabla:

Vamos a ver que pasa con (f,b)

4) (f,b) ?

f,0 → g
b,0 → c ?

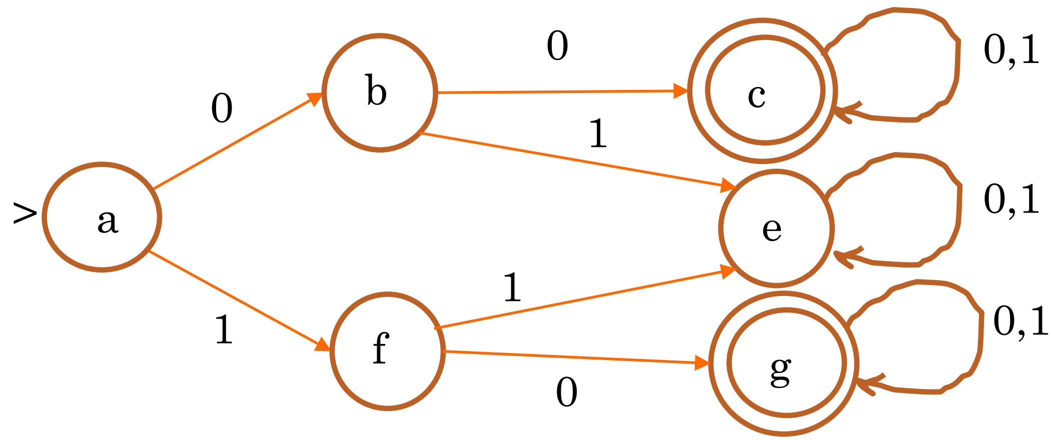
f,1 → e
b,1 → e ?

(f,b) es indistinguible, **no** lo marco en la tabla

Autómatas finitos. minimizacion

<u>Ejemplo 4:</u>

	a	b	c	e	f	g
a	X	X	X	X	X	X
b		X	X	X	X	X
c	//	//	X	X	X	X
e			//	X	X	X
f	///		//	///	X	X
g	//	//		//	//	X



Analizo el resto de la tabla:

Vamos a ver que pasa con (f,e)

4) (f,e) ? f,0 → g

e,0 → e

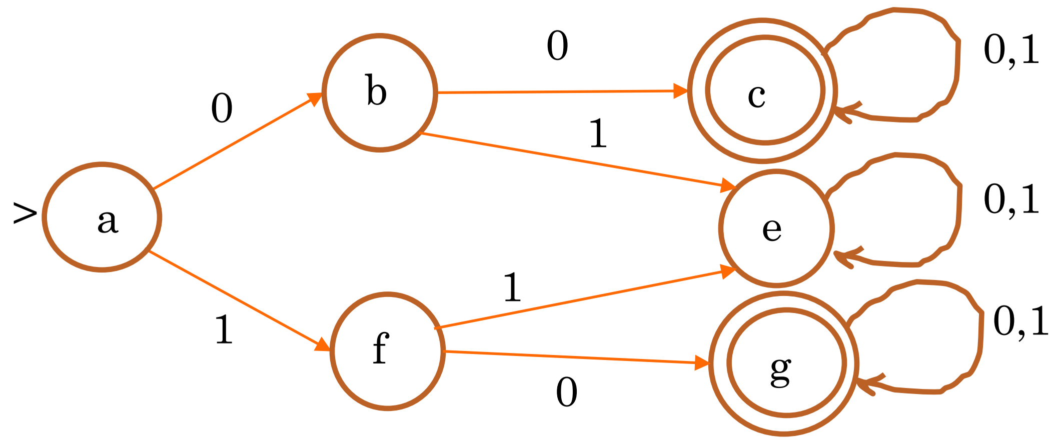
distinguible

(f,e)
es distinguible, lo marco en la tabla

AUTÓMATAS FINITOS. MINIMIZACION

Ejemplo 4:

	a	b	c	e	f	g
a	X	X	X	X	X	X
b		X	X	X	X	X
c	//	//	X	X	X	X
e	////		//	X	X	X
f	///		//	///	X	X
g	//	//		//	//	X



Analizo el resto de la tabla:

Vamos a ver que pasa con (e,a)

5) (e,a) ? e,0 → e

a,0 → b

?

e,1 → e

a,1 → f

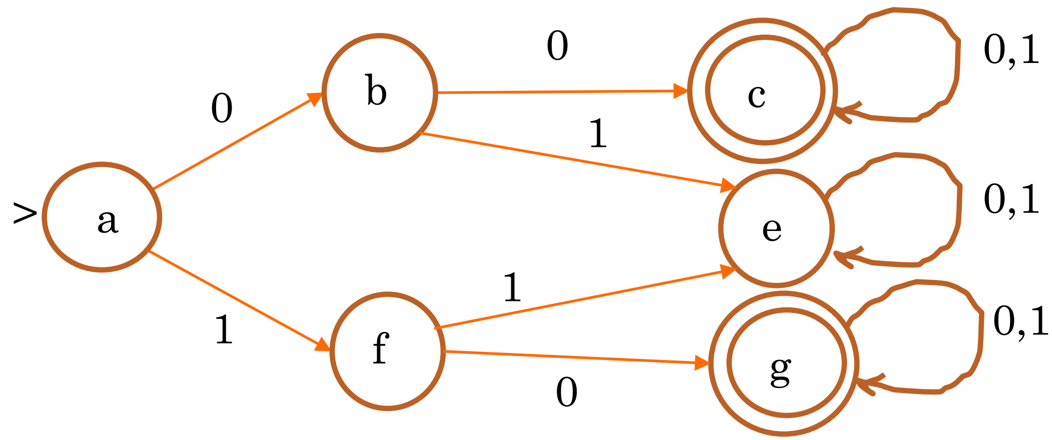
Es distinguible (está tachado en la tabla)

(e,a) es distinguible, lo marco en la tabla

Autómatas finitos. minimizacion

Ejemplo 4:

	a	b	c	e	f	g
a	X	X	X	X	X	X
b		X	X	X	X	X
c	//	//	X	X	X	X
e	////	////	//	X	X	X
f	///		//	///	X	X
g	//	//		//	//	X



Analizo el resto de la tabla:

Vamos a ver que pasa con (e,b)

5) (e,b) ? e,0 → e

b,0 → c

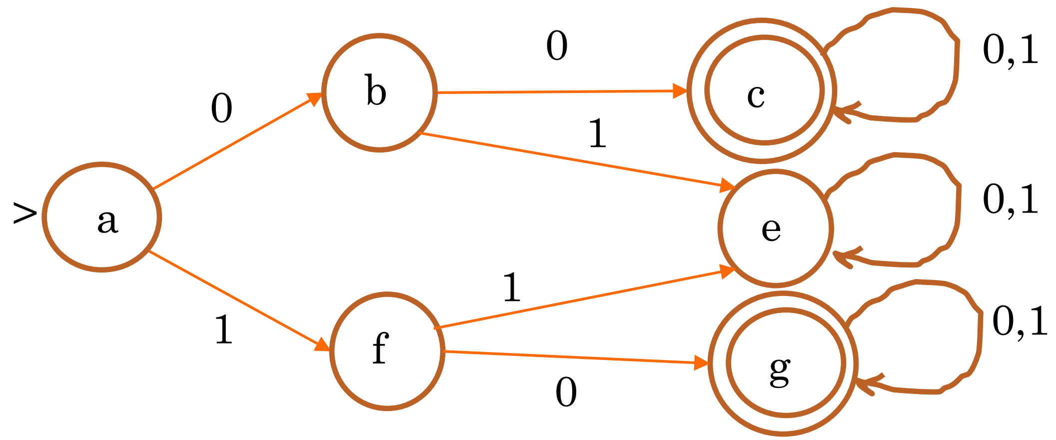
distinguible

(e,b)
es distinguible, lo marco en la tabla

AUTÓMATAS FINITOS. MINIMIZACION

Ejemplo 4:

/	a	b	c	e	f	g
a	X	X	X	X	X	X
b	V	X	X	X	X	X
c	//	//	X	X	X	X
e	////	////	//	X	X	X
f	///		//	///	X	X
g	//	//		//	//	X



Analizo el resto de la tabla:

Vamos a ver que pasa con (b,a)

6) (b,a) ? b,0 → c

a,0 → b

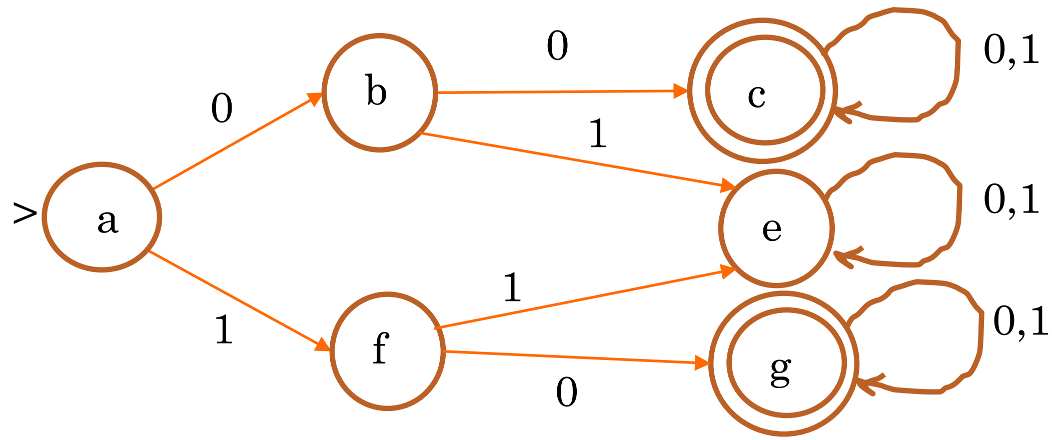
distinguible

(b,a)
es distinguible, lo marco en la tabla

Autómatas finitos. minimizacion

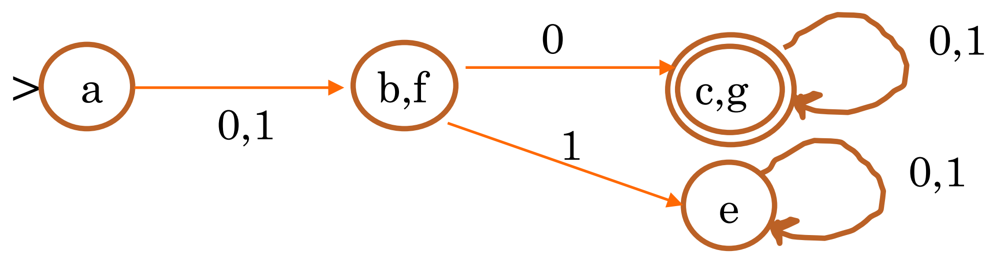
Ejemplo 4:

	a	b	c	e	f	g
a	X	X	X	X	X	X
b	V	X	X	X	X	X
c	//	//	X	X	X	X
e	////	////	//	X	X	X
f	///		//	///	X	X
g	//	//		//	//	X



Luego, se agrupan los estados indistinguibles.

Se reescribe el autómata considerando a (c,g) y a (b,f) como estados únicos.



Por último se renombra el autómata de esta forma: