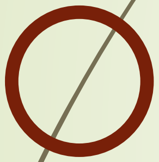
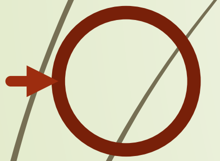
Expresiones Regulares y Lenguajes Regulares

Diagrama de transición de estados

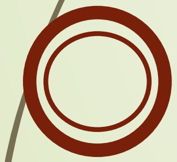
En los diagramas de estado podemos encontrar dos elementos: estados y transiciones.



Representación de un estado: son letras o símbolos



Representación de un estado inicial



Representación de un estado final o de aceptación

Transición: son arcos dirigidos que llevan asociadas etiquetas

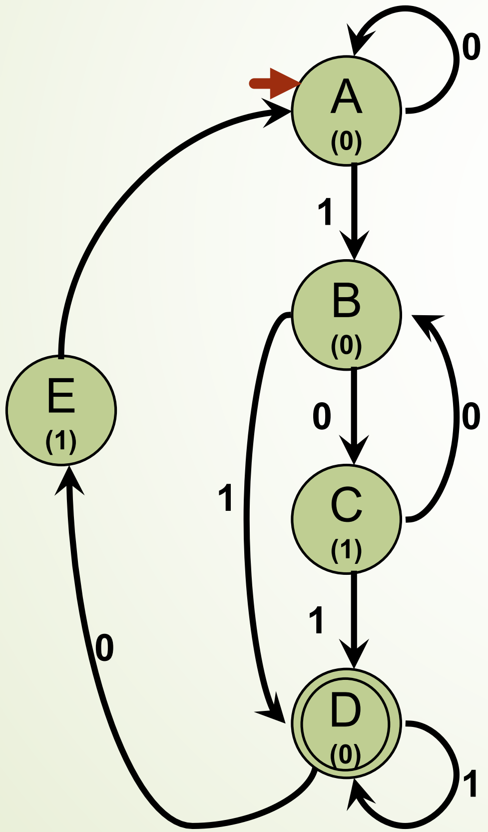
Diagrama de transición de estados

- **Ejemplos de Diagramas de Estado**

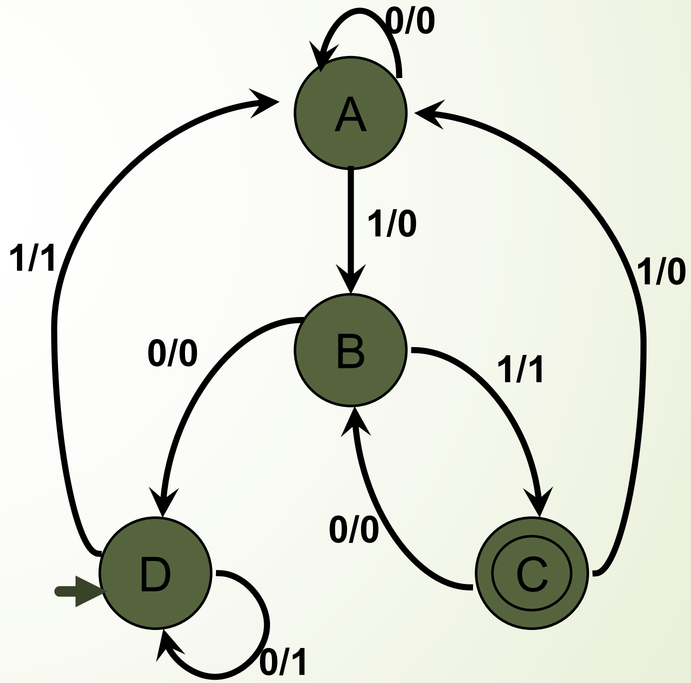
1 entrada

1 salida

5 estados



entrada/salida



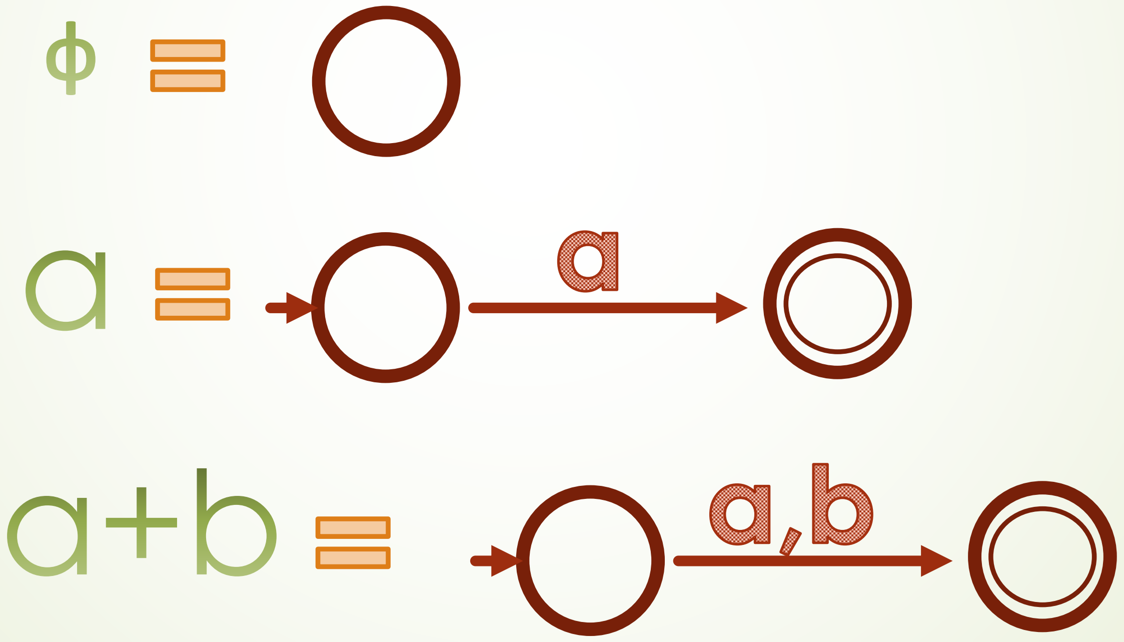
1 entrada

1 salida

4 estados

Representación de cadenas

Para realizar la representación de cadenas en un diagrama de estado se debe tener presente:

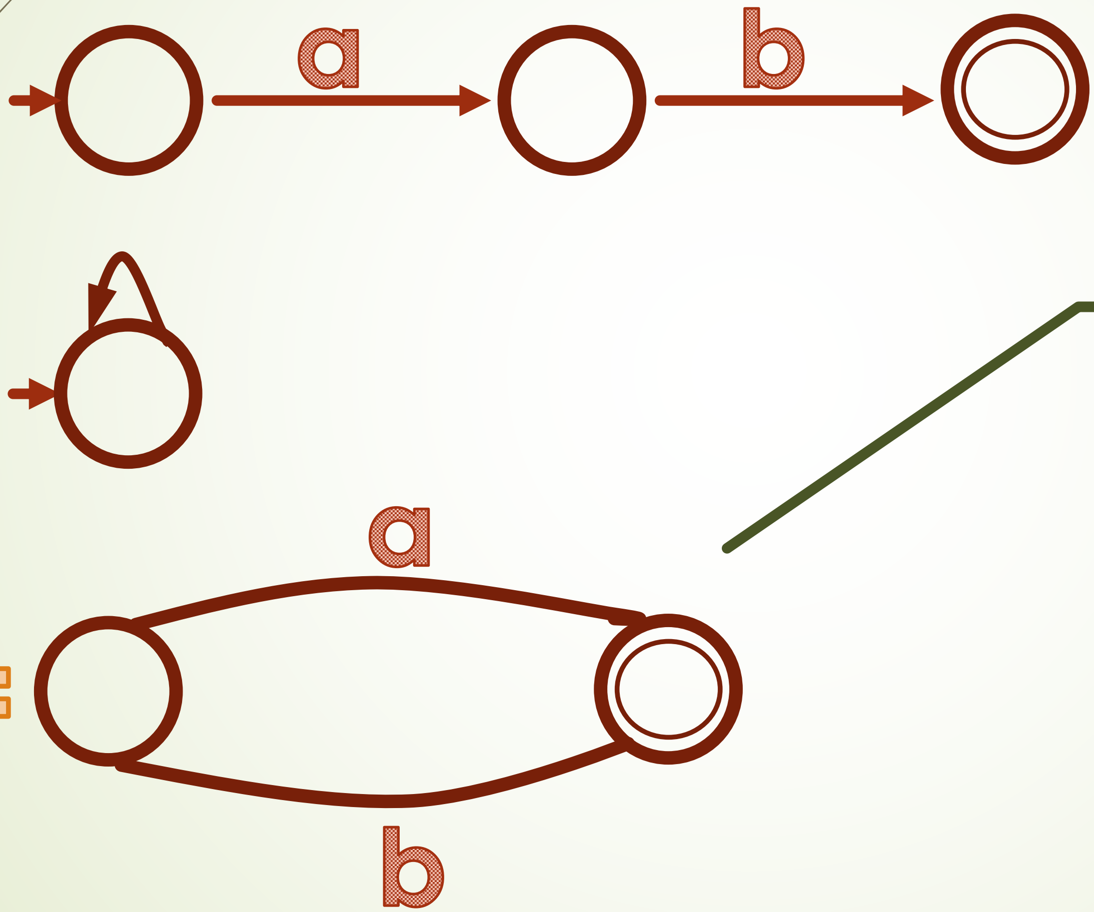


Representación de cadenas

a.b =

a* =

a | b =

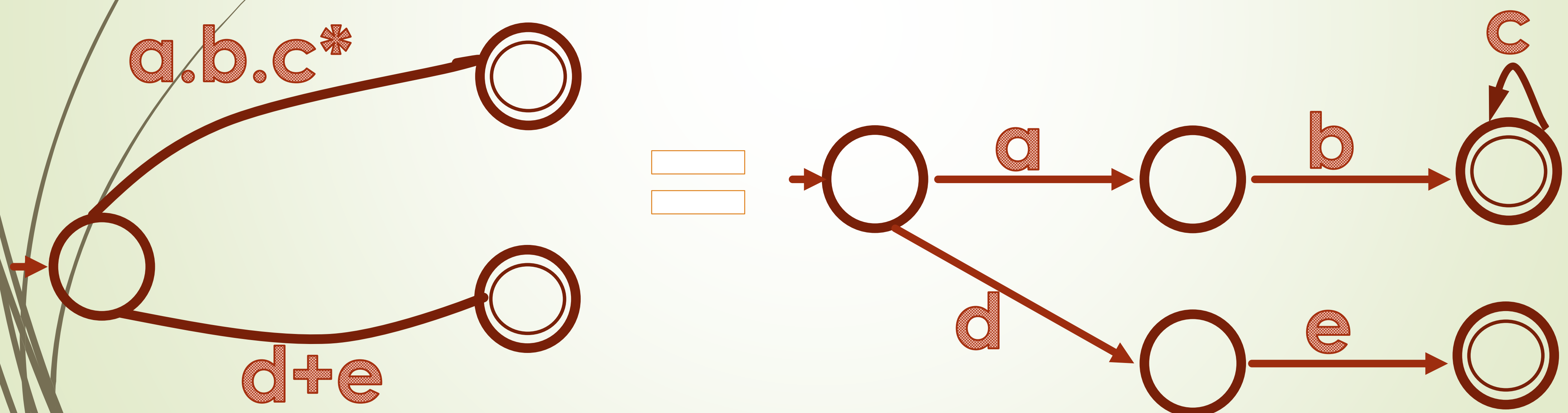


A | B = A o B
(A | B)* = Cero o muchos elementos de a o b
(A | B)⁺ = Uno a muchos elementos de a o b
[a-z] = Todo el alfabeto de a a z
[A - Z] = Todo el alfabeto de A a Z

Cadenas aceptadas

- **Ejemplo 1:** Expresión abc* | d+e

Esta expresión contiene dos términos separados por el operador lógico V, lo que lleva a que se de alguna de las dos condiciones: a.b.c* **o** d+e



Conversión de ER a FA

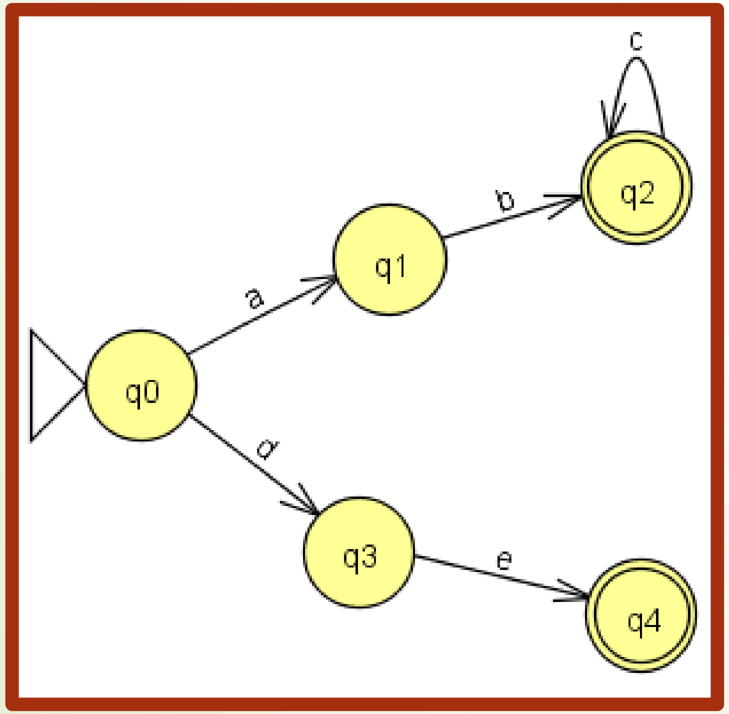
Las cadenas que acepta son:

Cadenas que empiezan por una única a, seguidas de una única b y en el estado final puede o no haber una o varias c

o

Por otro lado debe existir una única d seguida de una única e no acepta nada más

abc* | d+e



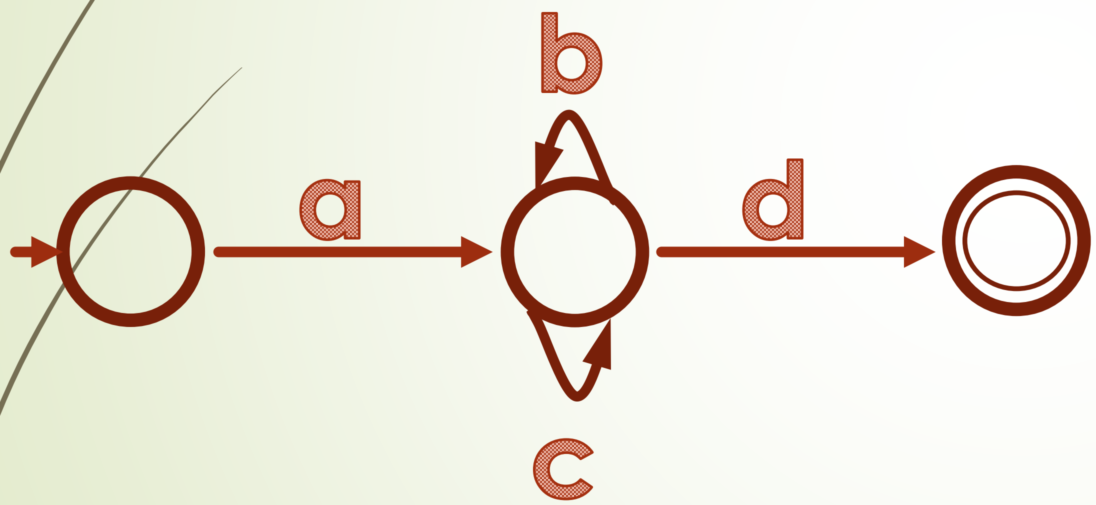
Aceptadas

ab	Accept
abc	Accept
abccc	Accept
abcccccc	Accept
de	Accept
a	Reject
b	Reject
d	Reject
e	Reject

No Aceptadas

Cadenas aceptadas

- **Ejemplo 2:** Expresión a(b|c)*d



La expresión tiene un estado **a**, *concatenado con*

(b|c)* = significa que puede o no existir una o varias **b o c,** *concatenado con*

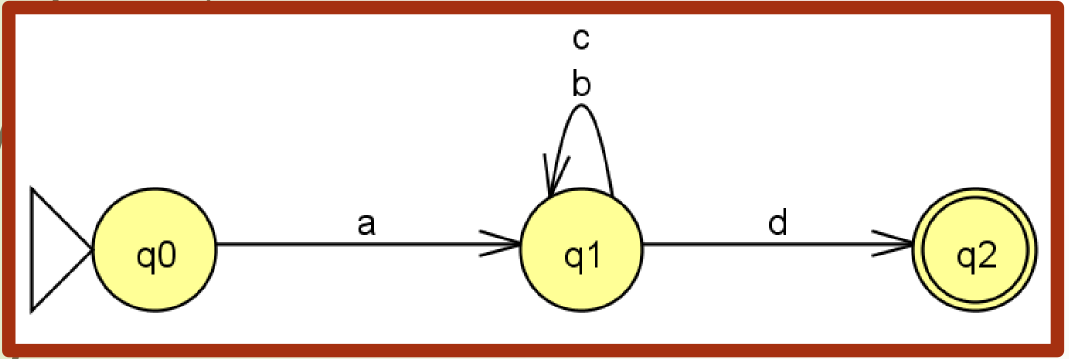
El estado terminal corresponde a una **d**

Conversión de ER a FA

$a(b|c)^*d$

Las cadenas que acepta son:

Cadenas que empiezan por una única a, seguida de una b o c (puede o no haber una o varias b o c) y termina con una única d.



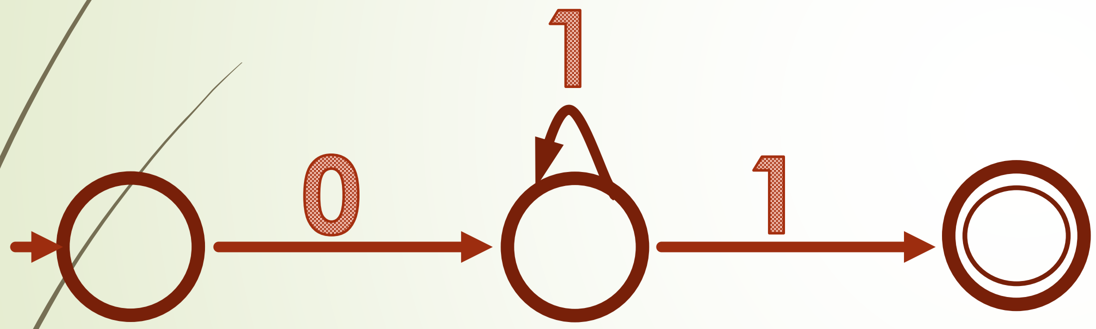
ad	Accept
abd	Accept
acd	Accept
abbd	Accept
abcd	Accept
abcbd	Accept
accbd	Accept
abbbcccd	Accept
accccccbd	Accept
ab	Reject
abbc	Reject
bbcd	Reject
cad	Reject

Aceptadas

No Aceptadas

Cadenas aceptadas

- **Ejemplo 3:** Expresión 01*+1



La expresión tiene un estado **0**, *concatenado con*

1* = significa que puede o no existir uno o varios **1**, *concatenado con*

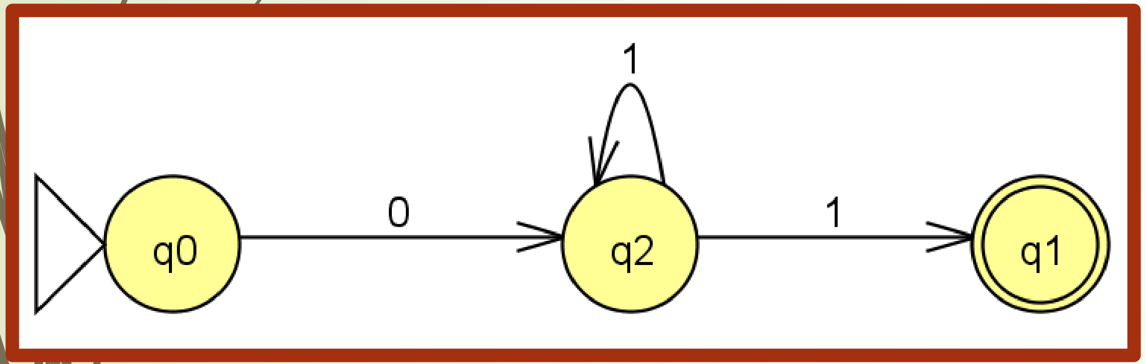
El estado terminal que corresponde a un **1**

Conversión de ER a FA

Las cadenas que acepta son:

Cadenas que empiezan por un único 0, seguida de un 1 (puede o no haber uno o más unos) y termina con un único 1.

01*+1



Aceptadas

01	Accept
011	Accept
0111	Accept
011111	Accept
010	Reject
0110	Reject
101	Reject
1110	Reject

No Aceptadas

Autómata Finito

- Es un modelo computacional que realiza cómputos en forma automática sobre una entrada para producir una salida.

- Este tiene una cantidad de memoria extremadamente limitada

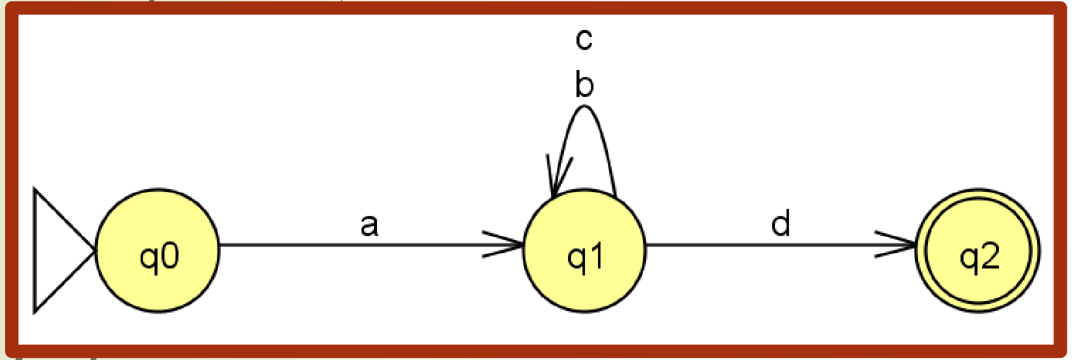
Definición formal de Autómata Finito

Formalmente, una máquina de estados finitos es una 5-tupla *(K, Σ, δ, s, F)* donde:

- *K* es un conjunto finito de estados;
- *Σ* es un alfabeto finito de símbolos de entrada;
- *s* es el estado inicial en *K*;
- *F* es el conjunto de estados finales o de aceptación y (evidentemente) subconjunto de *K*.
- *δ* es la relación de transiciones, que a partir de un estado y un símbolo del alfabeto obtiene un nuevo estado.

Ejemplo

- Teniendo en cuenta el ejemplo 2 anteriormente visto tenemos el siguiente autómata:



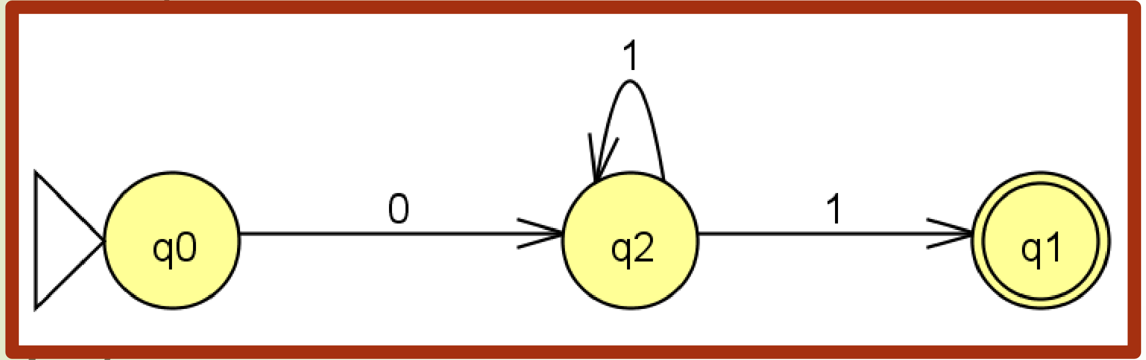
5-tupla *(K, Σ, δ, s, F)* donde:

M = ({q0, q1, q2} , {a, b, c, d} , δ, q0, {q2})
K **=***{ q0, q1, q2}*
Σ **=***{a, b, c, d)*
s **=** *q0*
F **=** *q2*
Donde la función *δ : {q0, q1, q2 } × {a, b, c, d} →*
{q0, q1, q2} viene dada por:
δ(q0, a) = q1
δ(q1, b) = q1
δ(q1, c) = q1
δ(q1, d) = q2

Ejemplo

- Teniendo en cuenta el ejemplo 3 anteriormente visto tenemos el siguiente autómata:

5-tupla *(K, Σ, δ, s, F)* donde:



M = ({q0, q1, q2} , {0, 1} , δ, q0, {q1})
K **=***{ q0, q1, q2}*
Σ **=***{0, 1)*
s **=** *q0*
F **=** *q1*
Donde la función *δ* : *{q0, q1, q2 }* × *{0, 1}* → *{q0, q1, q2}* viene dada por:
δ(q0, 0) = q2
δ(q2, 1) = q2
δ(q2, 1) = q1

Tabla de transición de estados

- Fila encabezada por los estados
- Columna encabezada por los símbolos de entrada

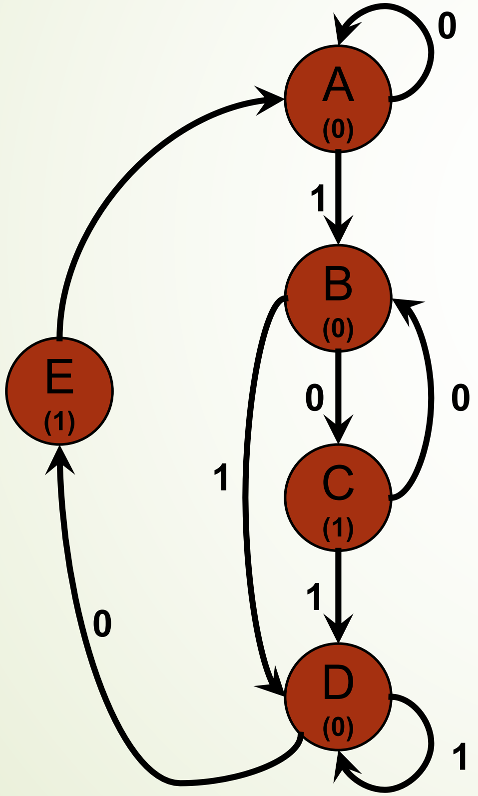
Símbolos de Entrada

	e_1	e_2	…	e_n
→ q_1		$f(q_1, e_2)$		
…				
$*q_m$				

Estados

Tabla de transición de estados ejemplo 1

- Ejemplo de Tabla de Estado



ESTADO ACTUAL	ESTADO SIGUIENTE		SALIDA
	0	1	
A	A	B	0
B	C	D	0
C	B	D	1
D	E	D	0
E	A	A	1

Tabla de transición de estados del ejemplo 2

Las transiciones son las siguientes:

$\delta(q0, a) = q1$

$\delta(q1, b) = q1$

$\delta(q1, c) = q1$

$\delta(q1, d) = q2$

Estado Inicial

Estado Final

	a	b	c	d
→ Q0	q1	∅	∅	∅
Q1	∅	q1	q1	q2
# Q2	∅	∅	∅	∅

Nota: La tabla no puede quedar vacía, en las transiciones que no existan, se deben llenar con el símbolo de vacío

Tabla de transición de estados del ejemplo 3

Las transiciones son las siguientes:

$\delta(q0, 0) = q2$
$\delta(q2, 1) = q2$
$\delta(q2, 1) = q1$

Estado Inicial

	0	1
→ Q0	q2	∅
Q2	∅	Q2, Q1
# Q1	∅	∅

Estado Final

Nota: La tabla no puede quedar vacía, en las transiciones que no existan, se deben llenar con el símbolo de vacío

Gracias