Matemáticas Discretas

Capítulo 8:
Circuitos Secuenciales, Máquinas y Autómatas de Estado Finito

Circuitos secuenciales

- Estudiaremos los sistemas donde la salida en un instante dado depende, no sólo de la entrada en ese mismo instante, sino del estado del sistema en el momento en que se introduce la entrada.
- Estos sistemas tienen memoria y se les llama circuitos secuenciales.
- En este tipo de circuitos, el estado interno del sistema depende del estado precedente de éste y de la entrada precedente.
- Las operaciones dentro de una computadora digital se realizan a intervalos discretos de tiempo.
- Una forma sencilla de introducir la secuenciación en los circuitos consiste en utilizar un retraso unitario de tiempo.

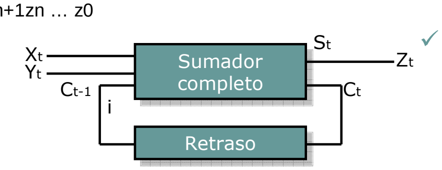
Retraso Unitario de Tiempo

- Un retraso unitario de tiempo acepta como entrada un bit x_t en el instante y tiene como salida x_{t-1}.

- Como ejemplo del retraso unitario de tiempo, analizaremos el **sumador en serie.**
- El sumador en serie es una aplicación del retraso unitario

Sumador en serie

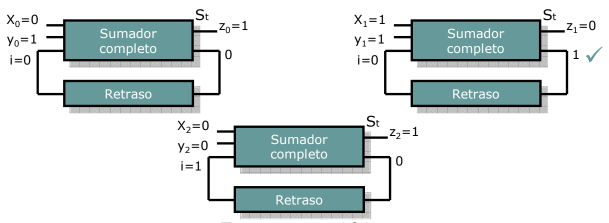
- Un sumador en serie acepta como entrada dos números binarios (x,y) y produce como salida la suma (z). Los dos números se introducen de manera secuencial por pares (x0,y0;…;xn,yn).
  - X = 0xnxn-1…x0
  - Y = 0ynyn-1…y0
  - Z = zn+1zn … z0



✓ El ingreso de dos números de entrada X y Y, tiene una salida Z

Sumador en serie

- Cálculo de la suma de dos números x y y mediante el sumador de serie. (x=010; y=011).
- El retraso unitario es el acarreo del sumador de serie



$Z = z_0 z_1 z_2 = 101$

Maquina de Estado Finito

- **Una Máquina de Estado finito es un modelo Abstracto de una máquina con una memoria interna primitiva.**
- **Definición:**
  - Una máquina de estado finita M consta de:
    - Un conjunto finito *I* de símbolos de entrada.
      - **{ a, b, c}**
    - Un conjunto finito *O* de símbolos de salida.
      - **{ 0, 1}**
    - Un conjunto finito S de estados.
      - **{ σ1, σ2}**
    - Una función de estado siguiente f de SxI en S.
      - **f (S, I) = S**
    - Una función de salida g de SxI en O.
      - **f (S, I) = O**
    - Un estado inicial σ Є S.

Escribimos M = (*I*, *O*, *S* , f, g, σ)

Cont…

- Sea I = {a,b}, O = {0,1} y S = {σ0, σ1}.
- I es la entrada, O es la salida y S es el estado.
- Definimos el par de funciones f:SxI→S y g:SxI → O, mediante las reglas de la tabla dada.

Tabla:

	f		g	
I / *S*	*a*	*b*	*a*	*b*
σ_0	σ_0	σ_1	0	1
σ_1	σ_1	σ_1	1	0

- **Entonces M = (I, O, S, f, g, σo)**

Cont…

Interpretación de la Tabla:

	f		g	
I / S	a	b	a	b
σ_0	σ_0	σ_1	0	1
σ_1	σ_1	σ_1	1	0

Para M = (*I*, *O*, *S* , f, g, σ)

$f(\sigma_0,a)=\sigma_0$ $g(\sigma_0,a)=0$

$f(\sigma_0,b)=\sigma_1$ $g(\sigma_0,a)=1$

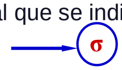
$f(\sigma_1,a)=\sigma_1$ $g(\sigma_0,a)=1$

$f(\sigma_1,b)=\sigma_1$ $g(\sigma_0,a)=1$

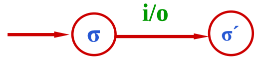
✓ σ_0 es elemento de la función de estado , a es elemento de la funcion de entrada y 1 es elemento de la función de salida

Diagramas de transición

- A este grupo de estados siguientes y de salida se los puede representar mediante un Diagrama de Transición
  - **Constan de:**
    - Los vértices que son los estados. 
    - Un estado inicial que se indica mediante una flecha.

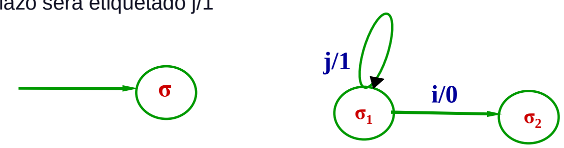


    - Si estamos en el estado "σ" y la entrada "i" produce una salida "o".
    - Y nos pasa al estado "σ´"; trazamos una arista dirigida del vértice "σ" hasta el vértice "σ´" y lo etiquetamos i/o.



Cont…

- **Definición:**
  - Sea M = (I, O, S, f, g, σ) una máquina de estado finito. El diagrama de transición de M es una digráfica G cuyos vértices son los miembros de de S. Una flecha indica el estado Inicial σ. Una arista dirigida (σ_1, σ_2) existe en G si existe una entrada *i* tal que f(σ_1, i) = σ2. En este caso, si g(σ_1, i)=O, la arista (σ_1, σ_2) se etiqueta i/o.
  - Si f(σ_1, j)= σ1 decimos que es un lazo dirigido en σ1, que también será etiquetado respecto a la función g; si g(σ_1, j)=1. El lazo será etiquetado j/1



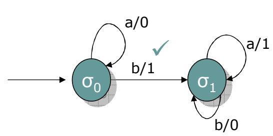
Cont…

- Realizar el diagrama de transición para la máquina de estado finito de la tabla dada.

I \ S	f: a	f: b	g: a	g: b
σ_0	σ_0	σ_1	0	1
σ_1	σ_1	σ_1	1	0

Para M = (*I*, *O*, *S* , f, g, σ)

f(σ_0,a)=σ_0 g(σ_0,a)=0
f(σ_0,b)=σ_1 g(σ_0,a)=1
f(σ_1,a)=σ_1 g(σ_0,a)=1
f(σ_1,b)=σ_1 g(σ_0,a)=1



✓ Trazamos una arista dirigida de σ_0 a σ_1 y la etiquetamos, la parte superior es la entrada y la parte inferior es la salida

Cadena de Entrada

- **Definición:**
  - Sea M = (I, O, S, f, g, σ) una máquina de estado finito. Una cadena de entrada para M es una cadena en I. La cadena

$$y_1 \ldots y_n$$

es la cadena de salida para M correspondiente a la cadena de entrada

$$\sigma = x_1 \ldots x_n$$

si existen estados σ_0, … , σ_n tales que

- σ0 = σ
- σi = f(σi-1 , xi) para i=1, … , n
- yi = g(σi-1 , xi) para i=1, … , n

Cont…

- **Ejemplo:**
  - Determinar la cadena de salida correspondiente a la cadena de entrada
    aababba
  - para la máquina de estado finito del ejemplo anterior

S \ I	f: a	f: b	g: a	g: b
σ_0	σ_0	σ_1	0	1
σ_1	σ_1	σ_1	1	0

$f(\sigma_0, a) = \sigma_0 \quad g(\sigma_0, a) = 0$

$f(\sigma_0, a) = \sigma_0 \quad g(\sigma_0, a) = 0$

$f(\sigma_0, b) = \sigma_1 \quad g(\sigma_0, b) = 1$

$f(\sigma_1, a) = \sigma_1 \quad g(\sigma_1, a) = 1$

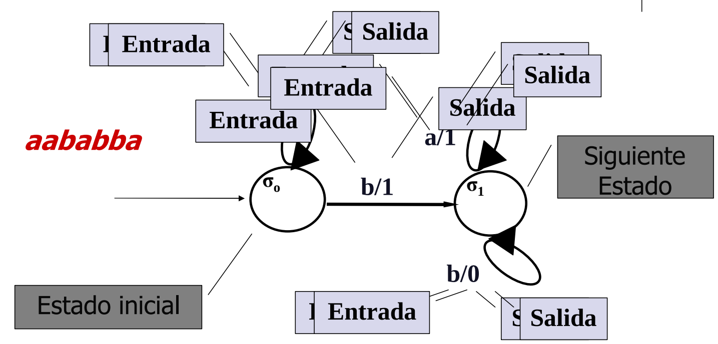
$f(\sigma_1, b) = \sigma_1 \quad g(\sigma_1, b) = 0$

$f(\sigma_1, b) = \sigma_1 \quad g(\sigma_1, b) = 0$

$f(\sigma_1, a) = \sigma_1 \quad g(\sigma_1, a) = 1$

La cadena de salida es 0011001

Resolución usando el diagrama de transición



0 0 1 1 0 0 1

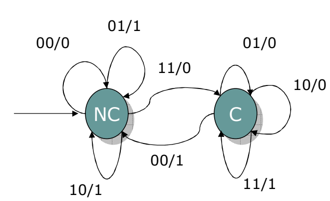
Cont…

- **Ejemplo:**
- **Una máquina de estado finito para el sumador en serie**
  - Diseñar una máquina de estado finto que realice la suma en serie.
  - Representaremos la máquina de estado finito mediante su diagrama de transición.
  - Como el sumador en serie acepta pares de bits ,
    - el conjunto de entrada será {00, 01, 10, 11}
    - El conjunto de salida es {0, 1}
  - Dada una entrada xy, realizamos una de las dos acciones siguientes:
    - sumamos **x** y **y**, o
    - sumamos **x, y** y **1,**

    según si el bit de acarreo sea 0 o 1.
  - Los estados serán **C**(acarreo) y **NC**(sin acarreo). Podemos ahora dibujar los vértices y designar el estado inicial en nuestro diagrama.

Cont…

- El conjunto de salida es: {0,1}.
- Tendremos dos entradas x *y* y.
- Realizamos dos acciones: sumamos x y y o x,y *y* 1.
- Tenemos dos estados C *(acarreo)* y NC *(sin acarreo).*
- NC es el estado inicial.



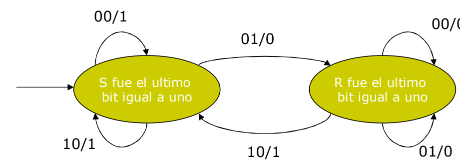
Cont…

- **Ejemplo:**
- **flip – flop SR**
  - Un flip-flop es un componente básico de los circuitos digitales, sirve como una celda de memoria de un bit.
  - El flip – flop SR se puede definir mediante la siguiente tabla:

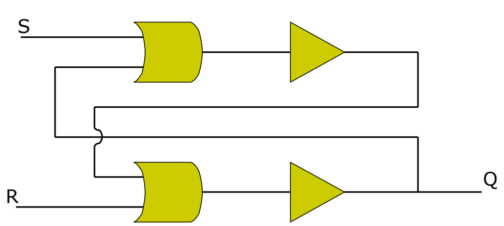
S	R	Q
1	1	No permitida
1	0	1 S
0	1	0 R
0	0	{1 si S fue ultimo bit igual a 1 {0 si R fue ultimo bit igual a 1

Cont…

- Podemos modelar el flip – flop SR como una **máquina de estado finito**, definiendo los estados "S fue el último bit igual a 1" y "R fue el último bit igual a 1".
- La entrada serán los valores de S y R. Definimos a Q como la salida. De manera arbitraria hemos escogido a S fue el último bit igual a 1" como estado inicial.



Cont…



✓ Implantación del flip-flop SR mediante un circuito secuencial

Cont…

- **Ejemplo:**
  - En una estación del Metro una máquina distribuye tiquetes sencillos a $600 pesos el tiquete. La máquina acepta monedas de $100, $200, $500, $1000. Mediante una tabla, describa los diferentes estados de la máquina y la salida.
- **Solución.**
  - Se supondrá que la máquina se encuentra en el estado e_0 perteneciente a E en el tiempo t_0. Al introducir una moneda en el tiempo t_i la salida será $g(x,e_s)$ donde e_s es el estado de la máquina en el tiempo t_i. A esta salida le sigue una transición de la máquina en el tiempo t_{i+1}dado por f(x,es).
  - Los estados del conjunto E serán:
    - e_0 = Estado inicial de la máquina sin introducir monedas.
    - e_1 = La máquina recuerda la inserción de $100.
    - e_2 = La máquina recuerda la inserción de $200.
    - e_3 = La máquina recuerda la inserción de $300.
    - e_4 = La máquina recuerda la inserción de $400.
    - e_5 = La máquina recuerda la inserción de $500.
    - e_6 = La máquina recuerda la inserción de $600 o más pesos.

Cont…

- **Cont…**
  - La función f: E x A ® E donde A = { n, 100, 200, 500, 1000, b } es la entrada donde n detalla el hecho de no introducir monedas y b hundir botón para obtener el tiquete, se detalla en la siguiente tabla:

	f					
	n	**100**	**200**	**500**	**1000**	***b***
$\mathbf{e_0}$	e_0	e_1	e_2	e_5	e_6	e_0
$\mathbf{e_1}$	e_1	e_2	e_3	e_6	e_6	e_1
$\mathbf{e_2}$	e_2	e_3	e_4	e_6	e_6	e_2
$\mathbf{e_3}$	e_3	e_4	e_5	e_6	e_6	e_3
$\mathbf{e_4}$	e_4	e_5	e_6	e_6	e_6	e_4
$\mathbf{e_5}$	e_5	e_6	e_6	e_6	e_6	e_5
$\mathbf{e_6}$	e_6	e_6	e_6	e_6	e_6	e_0

Cont…

- **Cont…**
  - En esta tabla por ejemplo, $f(e_0,500)=e_5$; lo que quiere decir que en el tiempo *t* siguiente la máquina recordará que se le han introducido $500.
  - $f(e_3,200)=e_5$, lo que significa que la máquina pasa del estado e_3; al estado e_5 ; lo que quiere decir que pasa de "recordar" que se le habrían introducido $300 a "recordar" que se le han introducido $500.
  - $f(e_5,200)=e_6$, lo que significa que la máquina pasa de "recordar" que se le habrían introducido $500 a "recordar" que se le han introducido más de $600, en este caso, la función de salida se diseñará para que devuelva $100 al comprador.
  - Al pulsar el botón, la máquina pasará al estado e0; si el estado actual es e_0 ó e_6.

Cont…

- **Cont…**
  - La función *g*:E x A ® B es la función de salida, que se detalla en el siguiente cuadro:

	g					
	n	**100**	**200**	**500**	**1000**	***b***
e0	*n*	*n*	*n*	*n*	400	*n*
e1	*n*	*n*	*n*	n	500	*n*
e2	*n*	*n*	*n*	100	600	*n*
e3	*n*	*n*	*n*	200	700	*n*
e4	*n*	*n*	*n*	300	800	*n*
e5	*n*	*n*	100	400	900	*n*
e6	*n*	100	200	500	1000	T

Cont…

- **Cont…**
  - En esta tabla, por ejemplo, $g(e_3,500) = 200$, lo que significa que la máquina pasa de "recordar" que se le habían introducido $300 a "recordar" $800 y por tanto devuelve $200. Como $f(e_3,500)=e_6$, la máquina pasa al estado e_6 y por último, como $g(e_6,b)=T$ recibe el tiquete.
  - Como $f(e_6,b)=e0$, la máquina retorna al estado inicial.
  - El conjunto de salida B será:
    - B = { *n,* 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000, T}
  - *n* significa que no hay salida.
  - T significa que se entrega el tiquete.

Cont…

- **Ejercicios:**
  1. Diseñe una máquina de estado finito que produzca la salida 1 si la entrada son *k* unos, donde *k* es múltiplo de 3; produce la salida 0 en caso contrario.
  2. Diseñe una máquina de estado finito que produzca la salida 1 cuando ve el primer 0 y hasta ver otro; a partir de ese momento produce la salida 0; en los demás casos, produce la salida 0.
  3. Dibuje el diagrama de transición de la máquina M={ A, B, E, *f*, *g*} donde A = {*a*, *b*}; B = {0, 1}; E = {e0, e1} dadas las tablas siguientes de *f* y *g.*

	f				*g*	
	a	*b*			a	b
e_0	e_0	e_1		e_0	1	1
e_1	e_1	e_1		e_1	0	1

Autómatas de Estado Finito

- Un autómata de estado finito es un tipo particular de máquina de estado finito.
- Los autómatas de estado finito tienen un interés especial, debido a su relación con los lenguajes.
- **Definición:**
  - Un **autómata de estado finito** A = (I, O, S, f, g, σ) es una máquina de estado finito en la que el conjunto de símbolos de salida es {0,1} y donde el estado actual determina la última salida. Aquellos estados para los cuales la última salida es 1 son los estados de aceptación.

Cont…

- Los **diagramas de transición de un autómata de estado finito** se dibujan con los estados de aceptación encerrados en círculos dobles y omitiendo los símbolos de salida.
- Esto se debe a que solo existen dos símbolos de salida {0,1} y cuando el último símbolo de salida es 1, se conoce como un estado de aceptación, caso contrario no lo es por lo que no es necesario señalar el símbolo de salida en los lazos dirigidos.

Cont…

- El estado inicial es **σ0**. Mostrar que *A* es un autómata de estado finito y determinar el conjunto de estados de aceptación.

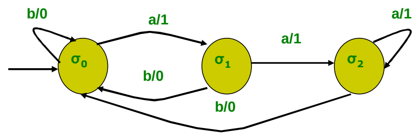
	f		g	
I \ S	a	b	a	b
σ_0	σ_1	σ_0	1	0
σ_1	σ_2	σ_0	1	0
σ_2	σ_2	σ_0	1	0

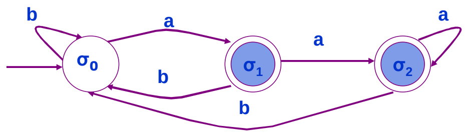
f(σ0,a)= σ1 g(σ0,a)= 1
f(σ0,b)= σ0 g(σ0,b)= 0
f(σ1,a)= σ2 g(σ1,a)= 1
f(σ1,b)= σ0 g(σ1,b)= 0
f(σ2,a)= σ2 g(σ2,a)= 1
f(σ2,b)= σ0 g(σ2,b)= 0

- Si estamos en el estado **σ0**, la última salida fue **0**. Si estamos en el estado **σ1** o el estado **σ2** la última salida fue **1**; así que ***A*** es un autómata de estado finito. Los estados de aceptación son **σ1** y **σ2**.

Diagrama de transición de un Autómata de EF

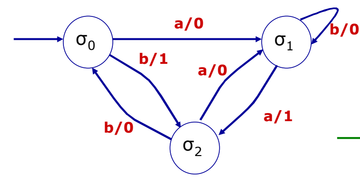
- $A=(I,S,f,A,\sigma)$.
- $I=\{ a,b \}$
- $S=\{\sigma_0, \sigma_1, \sigma_2 \}$
- $A = \{\sigma_2 \}$: $\sigma= \sigma_0$

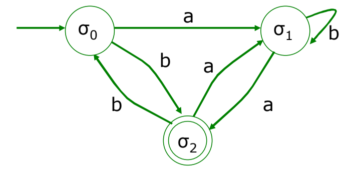




Cont…

- **Ejemplo:**
  - A partir del diagrama de autómata de estado finito dibujar el diagrama de transición de una máquina de estado finito.





Autómatas de estado finito aceptados

- **Definición**
  - Sea $A=(I, S, f, A, \sigma)$ un A.E.F.
  - Sea $x = x_1\ldots x_n$ un arreglo no nulo de I. Si existen estados $\sigma_0,\ldots, \sigma n$ que satisfacen:

    (a) $\sigma_0 = \sigma^*$

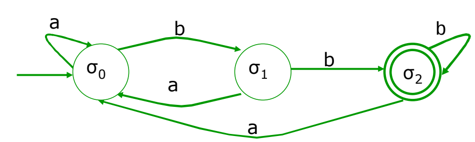
    (b) $f(\sigma_{i-1}, x_1) = \sigma_i$ para $i = 1,\ldots,n$;

    se dice que x es aceptado por A.
  - Al conjunto de arreglos aceptados por A se denota Ac(A)
  - Al camino (dirigido)($\sigma_0,\ldots, \sigma_n$) se le llama camino que representa a α en A.

Cont…

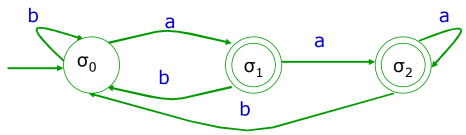
- **Ejemplo:**
  - **A = (I, O, S, f, g, σ) donde: I = {a,b}; S= {σ_0, σ_1, σ_2}; A={σ_2}; $\sigma= \sigma_0$ y f está dada por la siguiente tabla:**
  - Si una cadena se utiliza como entrada de un autómata de estado finito, terminaremos en un estado de aceptación o en uno de no aceptación. La situación de este estado final determina si la cadena es **aceptada** por el autómata de estado finito.

	f	
I \ S	a	b
σ_0	σ_0	σ_1
σ_1	σ_0	σ_2
σ_2	σ_0	σ_2



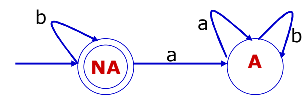
Cont…

- **Ejemplo:**
  - Existe una cadena "abaa". Verificar si el autómata de estado finito mostrado a continuación acepta esta cadena.
  - Comenzamos en el estado σ0. Cuando a es la entrada, pasamos al estado σ1. Cuando b es la entrada, vamos al estado σo. Cuando a es la entrada, pasamos a estado σ1. Por último, cuando se utiliza como entrada el último símbolo a, pasamos al estado σ2. El camino (σo, σ1, σo, σ1, σ2) representa la cadena abaa. Como el estado final σ2 es un estado de aceptación, la cadena abaa es aceptada por el autómata de estado finito.



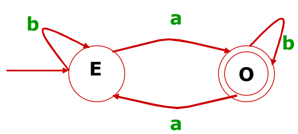
Cont…

- **Ejemplo:**
  - Diseñar un autómata de estado finito que acepte precisamente aquellas cadenas sobre {a,b} que no tengan letras a.
  - La idea es utilizar dos estados:
    - A: Se encontró una a.
    - NA: No se encontró una a.
  - El estado NA es el estado inicial y el único estado de aceptación. Por lo tanto el diagrama de transición del autómata de estado finito sería:



Cont…

- **Ejemplo:**
  - Diseñar un autómata de estado finito que acepte precisamente aquellas cadenas sobre {a,b} que contengan un número impar de letras a.
  - Los dos estados son:
    - E: Se encontró un número par de a.
    - O: Se encontró un número impar de a.
  - El estado E es el estado inicial y O es el único estado de aceptación. Su diagrama de transición es:



Cont…

- **Algoritmo del autómata de estado finito anterior**
  - Entrada: n, la longitud de la cadena $s_1,s_2,\ldots,s_n$
  - Salida:
    - "Aceptar" si la cadena es aceptada.
    - "Rechazar" si la cadena no es aceptada.

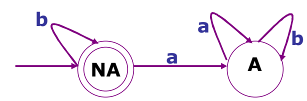
```
Procedure fsa(s,n)
   state:='E'
   for i:=1 to n do
      begin
      if state='E' and s_i='a' then
                   state:='O'
      if state='O' and s_i='a' then
                   state:='E'
      end
   if state='O' then
      return("Aceptar")
   else
      return("Rechazar")
end fsa
```

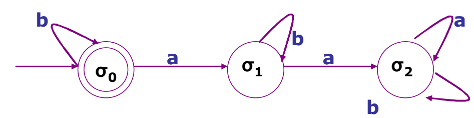
Autómatas Equivalente

- **Definición:**
  - Si dos autómatas de estado finito aceptan precisamente las mismas cadenas, decimos que los autómatas son **equivalentes.**
  - Los autómatas de estado finito A y A´ son equivalentes si Ac(A) = Ac(A´).
- Si definimos una relación R sobre un conjunto de autómatas de estado finito mediante la regla ARA'.
- Para que A y A' sean equivalentes, R debe ser una relación de equivalencia.
- Cada clase de equivalencia consta de un conjunto de autómatas de estado finito equivalentes entre sí.

Cont…

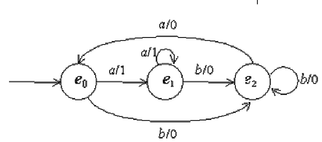
- **Ejemplo:**
  - **Podemos verificar con la cadena "abaab" que los siguientes diagramas son equivalentes:**



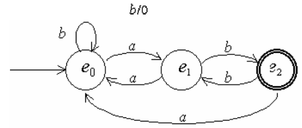


Cont…

- **Ejercicios:**
  - **Muestre que la máquina de estado finito descrita por la siguiente gráfica es un autómata de estado finito.**
    - **Cuál es el estado de aceptación?**



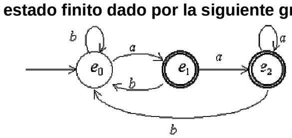
  - **Dada la gráfica del autómata de estado finito**
    - **Dibuje la tabla de transición de estados y la tabla de salida.**



  - **Determine si la cadena *abbaa* es aceptada por los autómatas de los dos ejercicios anteriores.**

Cont…

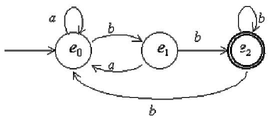
- **Ejercicios:**
  - **Trace la gráfica de un autómata de estado finito que acepte cadenas del conjunto A = { *a, b*} que posean:**
    - **Un número par de *a.***
    - **Al menos dos *a.***
    - **Exactamente dos *a.***
    - **Contiene *n* letras *a*, donde *n* es un número múltiplo de 3.**
  - **Dado A = { *a, b*}, muestre que una cadena de entrada es aceptada por el autómata de estado finito dado por la siguiente gráfica:**



    - **Sí y sólo sí la cadena termina en a. Cuál es la tabla de transición de estados?**

Cont…

- **Ejercicios:**
  - **Muestre que una cadena de entrada, dado A = { *a, b*} es aceptada por el autómata de estado finito dado por la siguiente gráfica:**



- **Sí y sólo sí la cadena termina en *bb*. Cuál es la tabla de transición de estados?**