

UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL VILLA MARÍA
INGENIERÍA EN SISTEMAS DE LA INFORMACIÓN
SINTAXIS Y SEMÁNTICA DE LOS LENGUAJES
TRABAJO PRÁCTICO N°2

Docentes:

- Titular Ing. Rinaldi, Mario
- J.T.P Doc. Palombarini, Jorge

Integrantes del grupo:

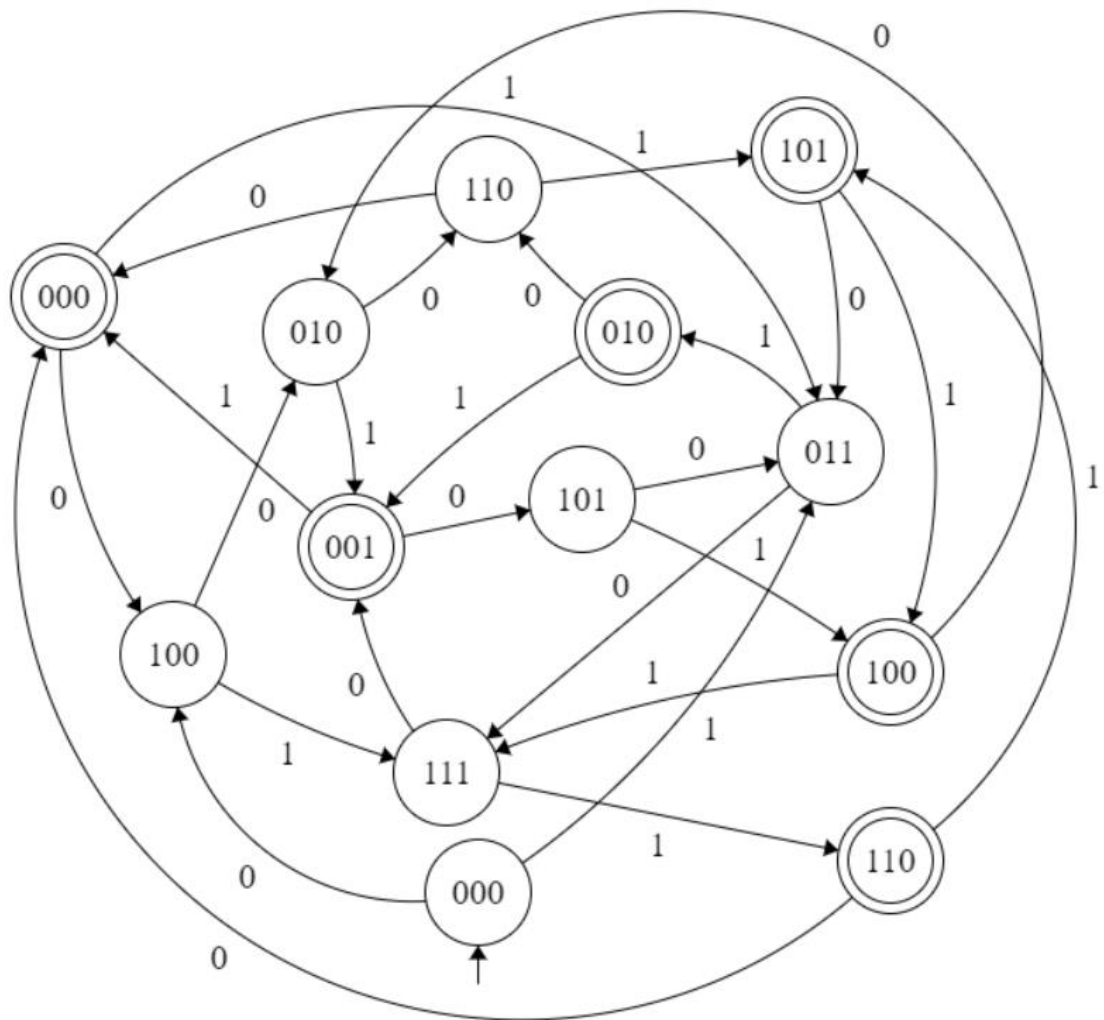
- Bartoloni, Agustín - 13027 – semiagustin10@gmail.com
- Sanchez, Federico – 13614 – federicosan140200@gmail.com
- Simonin, Eloy – 13727 – simonineloy123@gmail.com

Fecha de entrega: 30/09/2020

1.


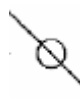
a.

Parte 1.



Notas:

Los estados están de tal forma: $x_1x_2x_3$.

El estado 0 de los niveladores es  mientras que el estado 1 .

La transición 0 es una bolita ingresando en A, la transición 1 es una bolita ingresando por B.

Parte 2.

a)

```
1 from automata.fa.dfa import DFA
2 from random import randint
3 dfa = DFA(
4     #x1x2x3
5     states={'000', '010', '011', '100', '101', '110', '111', '000f', '001f', '010f', '100f', '101f', '110f'}, # 0 -> por A | 1 -> por B
6     input_symbols={'0', '1'},
7     transitions={
8         '000': {'0': '100', '1': '011'},
9         '010': {'0': '110', '1': '001f'},
10        '011': {'0': '111', '1': '010f'},
11        '100': {'0': '010', '1': '111'},
12        '101': {'0': '011', '1': '100f'},
13        '110': {'0': '000f', '1': '101f'},
14        '111': {'0': '001f', '1': '110f'},
15        '000f': {'0': '100', '1': '011'},
16        '001f': {'0': '101', '1': '000f'},
17        '010f': {'0': '110', '1': '001f'},
18        '100f': {'0': '010', '1': '111'},
19        '101f': {'0': '011', '1': '100f'},
20        '110f': {'0': '000f', '1': '101f'},
21    },
22    initial_state='000',
23    final_states={'000f', '001f', '010f', '100f', '101f', '110f'} # estados terminados en f -> estados finales
24 )
25
26 print("La validación es: ", dfa.validate()) # Punto a
```

La validación es: True

Código

Salida del programa

Nota: Los estados terminados en f representan a los estados finales.

b)

```
28 # Punto b
29 Lista_Cadenas = ["0", "1", "00", "01", "10", "11", "101", "010", "111", "1111", "1010", "0100", "0011", "0001", "1110110", "1100110", "0100101110010", "0000110111111"]
30 númeroAleatorio = randint(0,19)
31 if dfa.accepts_input(Lista_Cadenas[númeroAleatorio]):
32     print("La cadena ", Lista_Cadenas[númeroAleatorio], " es aceptada.")
33 else:
34     print("La cadena ", Lista_Cadenas[númeroAleatorio], " es rechazada.")
```

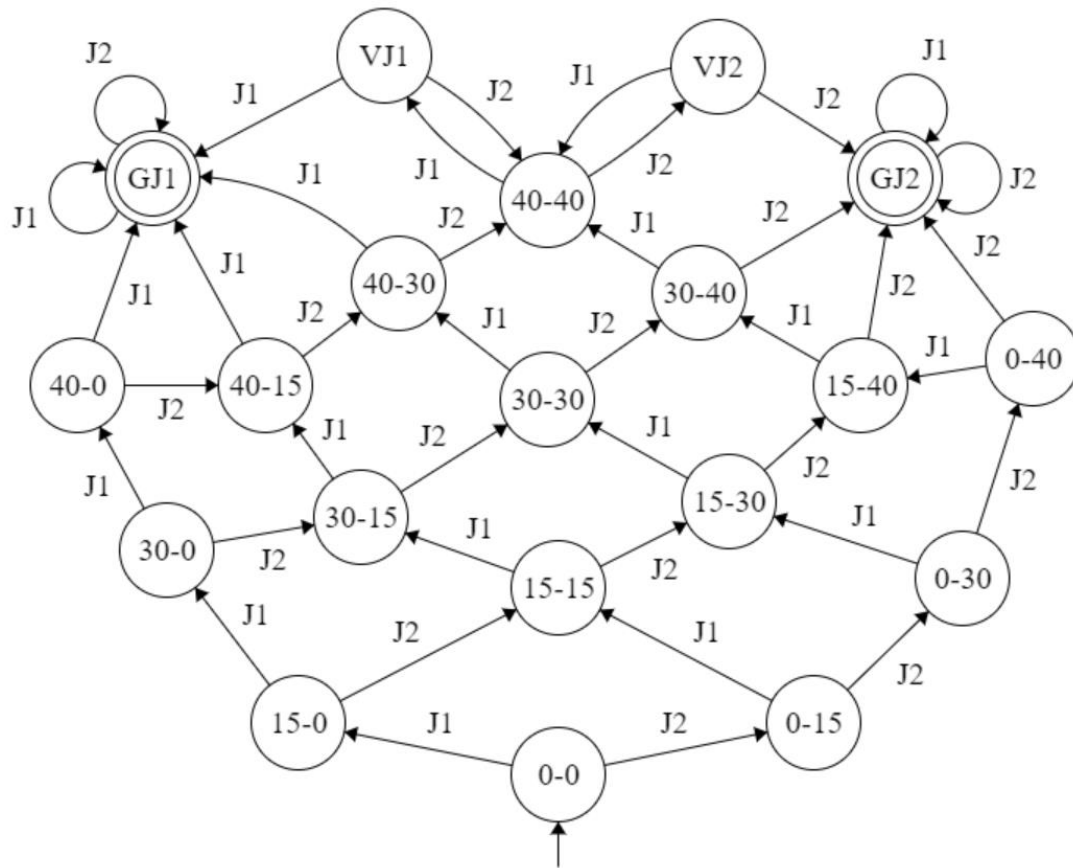
La cadena 1010 es aceptada.

La cadena 0100101110010 es rechazada.

La cadena 1011 es aceptada.

La cadena 00 es rechazada.

b.



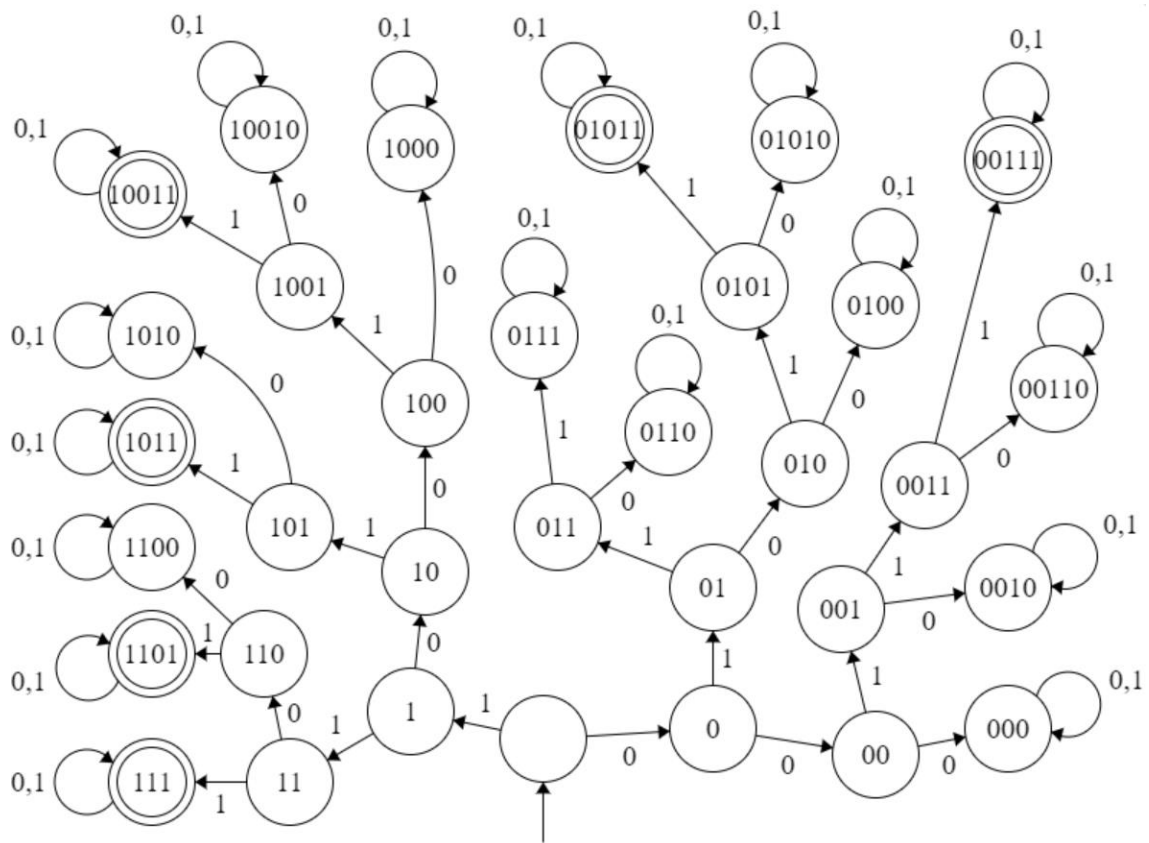
Notas:

1. Los estados están representados de la siguiente manera:

Puntajes del J1- Puntajes del J2

2. **J1:** transición en el que el jugador 1 hace un tanto.
3. **J2:** transición en el que el jugador 2 hace un tanto.
4. **VJ1:** estado en el que el jugador 1 tiene ventaja.
5. **VJ2:** estado en el que el jugador 2 tiene ventaja.
6. **GJ1:** estado final en el que el jugador 1 gana el game.
7. **GJ2:** estado final en el que el jugador 2 gana el game.

C.



Notas:

1 – Significa que el parcial está aprobado.

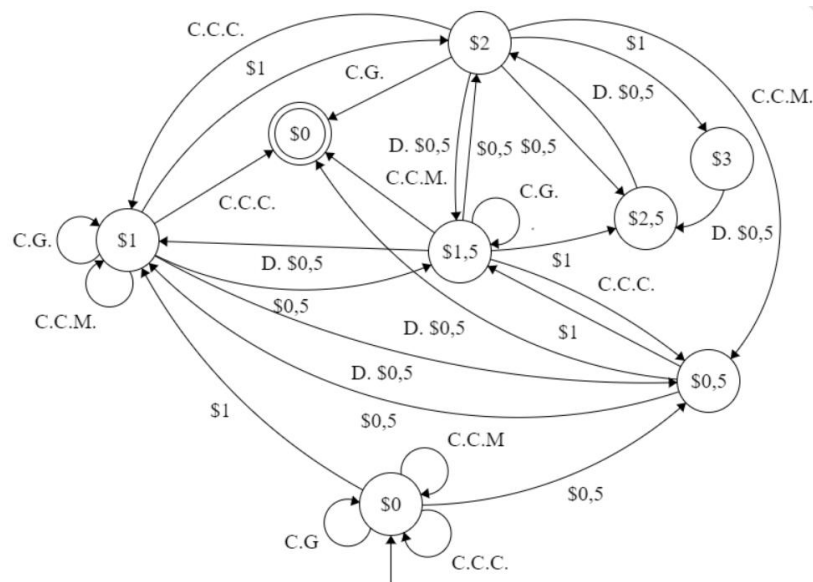
2 – Significa que el parcial está desaprobado.

Estados:

Parcial 1 Parcial 2 Parcial 3 Recuperatorio 1 Recuperatorio 2.

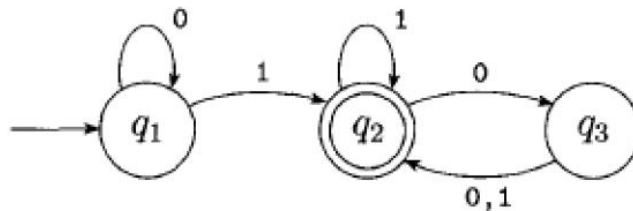
Como se cancela la operación?

d.



2.

Autómata 1.



Definición Formal:

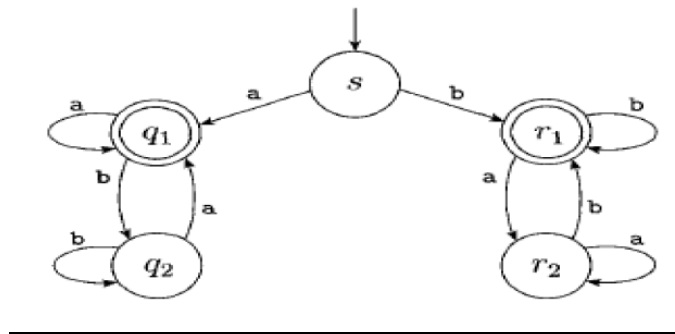
- Estados: $\{q1, q2, q3\}$
- Alfabeto: $\{0,1\}$
- Función transición:

	0	1
q1	q1	q2
q2	q3	q2
q3	q2	q2

- Estado inicial: $\{q1\}$ NO ES UN CONJUNTO
- Estado final: $\{q2\}$

El lenguaje regular que reconoce es: $\{w/w \text{ es una cadena que termina en } 1 \text{ o al menos dos } 0\}$ Al menos dos ceros no, ya que no acepta por ejemplo 1000

Autómata 2.



Definición Formal:

- Estados: $\{s, q1, q2, r1, r2\}$
- Alfabeto: $\{a, b\}$
- Función transición:

	a	b
s	q1	r1
q1	q1	q2
q2	q1	q2
r1	r2	r1
r2	r2	r1

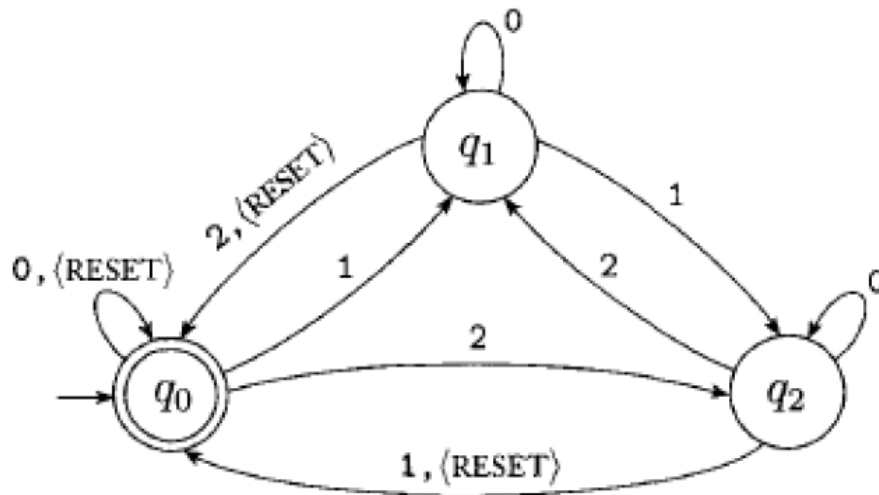
- Estado inicial: $\{s\}$
- Estado final: $\{q1, r1\}$

El lenguaje regular que reconoce es: $\{w/w \text{ es una cadena que empieza y termina con el mismo símbolo}\}$

Los ejemplos son:

- ba
- bb
- abba
- baab
- abbaababa

Autómata 3.



Definición Formal:

- Estados: $\{q_0, q_1, q_2\}$
- Alfabeto: $\{0, 1, 2, \text{RESET}\}$
- Función transición:

	0	1	2	RESET
q0	q0	q1	q2	q0
q1	q1	q2	q0	q0
q2	q2	q0	q1	q0

- Estado inicial: $\{q_0\}$
- Estado final: $\{q_0\}$

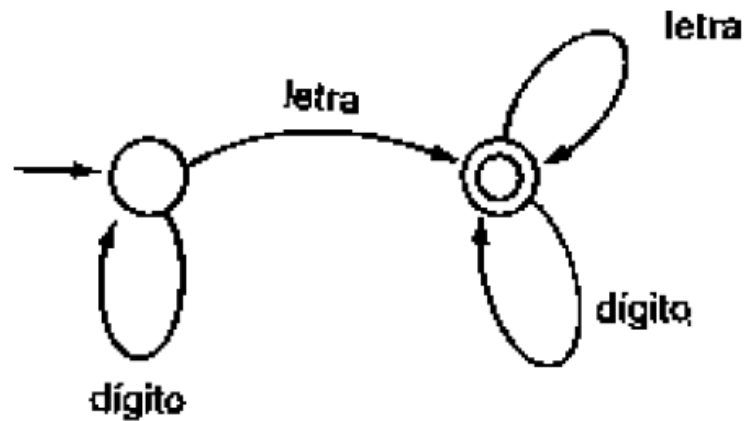
El lenguaje regular que reconoce es: $\{w/w \text{ es una cadena que contiene reset o que la suma de los números es múltiplo de tres.}\}$

Los ejemplos son:

- Reset
- 111
- 12
- 222
- 111111

Puede contener RESET y no ser aceptada. por ejemplo 1RESET11, etc. La suma de los números siempre es después del último RESET en caso de existir

Autómata 4.



Definición Formal:

- Estados: $\{q0, q1\}$
- Alfabeto: $\{\text{dígito}, \text{letra}\}$
- Función transición:

	dígito	letra
q0	q0	q1
q1	q1	q1

- Estado inicial: $\{q0\}$ *no es un conjunto*
- Estado final: $\{q1\}$

El lenguaje regular que reconoce es: $\{w/w \text{ es una cadena que contiene al menos una letra}\}$

Los ejemplos son:

- dígito dígito letra dígito dígito letra letra
- letra
- dígito dígito letra
- letra letra dígito
- dígito letra letra letra dígito dígito

Autómata 5.



Definición Formal:

- Estados: $\{q_0, q_1, q_2, q_3\}$
- Alfabeto: $\{a, b\}$
- Función transición:

	a	b
q0	q1	q0
q1	q2	q1
q2	q3	q2
q3	q3	q3

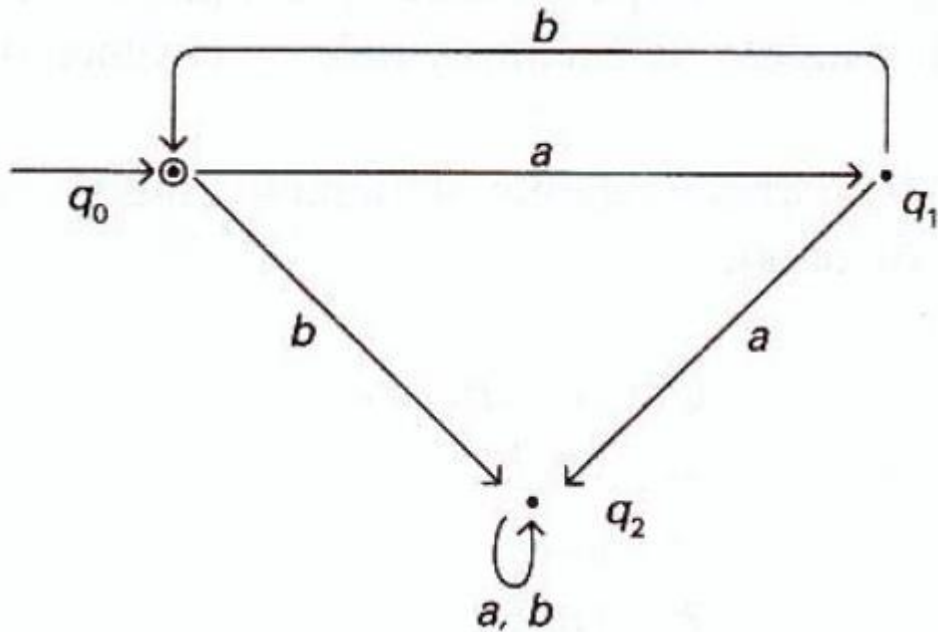
- Estado inicial: $\{q_0\}$ No es un conjunto
- Estado final: $\{q_2\}$

El lenguaje regular que reconoce es: $\{w/w \text{ es una cadena que contiene dos a y termina en b}\}$ No necesariamente, puede no terminar en b

Los ejemplos son:

- aab
- bbbaab
- baab
- bbaab
- bbbbbbbbaab

Autómata 6.



Definición Formal:

- Estados: $\{q_0, q_1, q_2\}$
- Alfabeto: $\{a, b\}$
- Función transición:

	a	b
q0	q1	q2
q1	q2	q0
q2	q2	q2

- Estado inicial: $\{q_0\}$ No es un conjunto
- Estado final: $\{q_0\}$

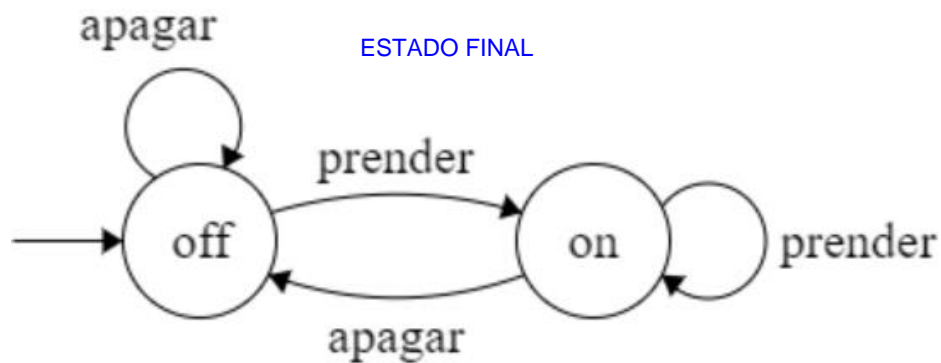
El lenguaje regular que reconoce es: $\{w/w \text{ es una cadena que contiene } n \text{ cantidad de veces la secuencia "ab"}\}$

Los ejemplos son:

- ab
- abab
- ababab
- abababab
- ababababab

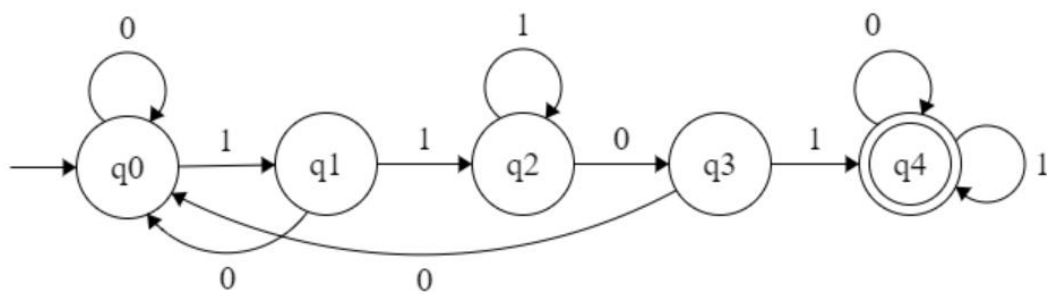
DESPUES DEL ÚLTIMO RESET

3. Es un autómata que representa una secuencia de dígitos que al sumarse da múltiplo de 3 o se resetea el valor sin importar el estado en el que esté. Los estados representan la suma de los números, siendo q0 un estado que siempre representa un múltiplo de 3.
- 4.



5. Parte 1

a.



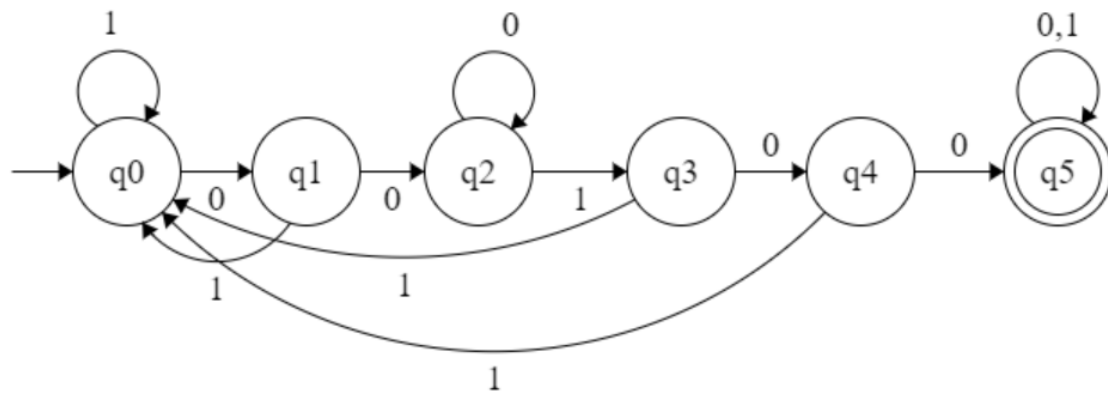
Definición Formal:

- Estados: {q0, q1, q2, q3, q4}
- Alfabeto: {0, 1}
- Función transición:

	0	1
q0	q0	q1
q1	q0	q2
q2	q3	q2
q3	q0	q4
q4	q4	q4

- Estado inicial: {q0} *no es un conjunto*
- Estado final: {q4}

b.



Definición Formal:

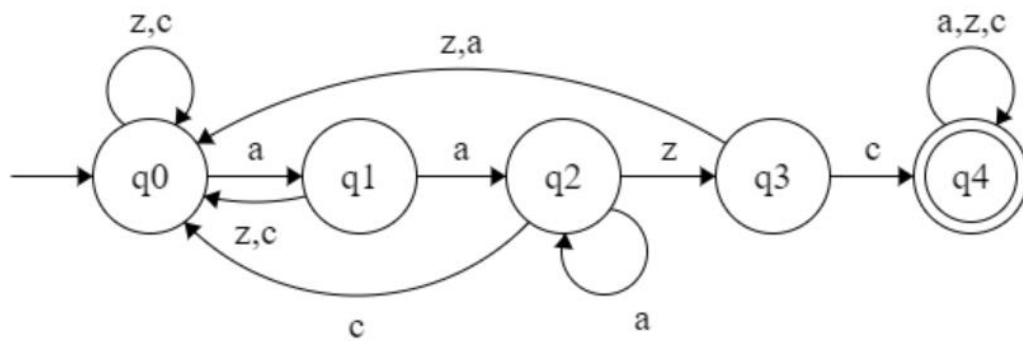
- Estados: {q0, q1, q2, q3, q4, q5}
- Alfabeto: {0, 1}
- Función transición:

	0	1
q0	q1	q0
q1	q2	q0
q2	q2	q3
q3	q4	q0
q4	q5	q0
q5	q5	q5

- Estado inicial: {q0} *no es conjunto*
- Estado final: {q5}

La transición a no vuelve a q0, ya que hace que el autómata no reconozca cadenas válidas como aazaazc, etc

C.



Definición Formal:

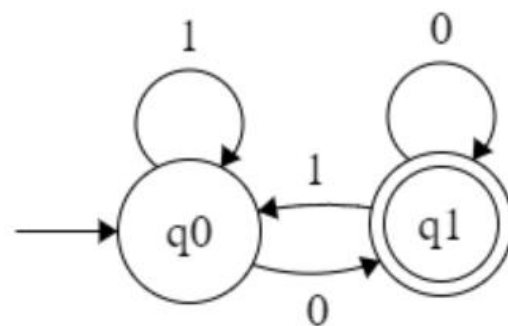
- Estados: {q0, q1, q2, q3, q4 }
- Alfabeto: {a, z, c}
- Función transición:

	a	z	c
q0	q1	q0	q0
q1	q2	q0	q0
q2	q2	q3	q0
q3	q0	q0	q4
q4	q4	q4	q4

- Estado inicial: {q0} no es conjunto
- Estado final: {q4}

d.

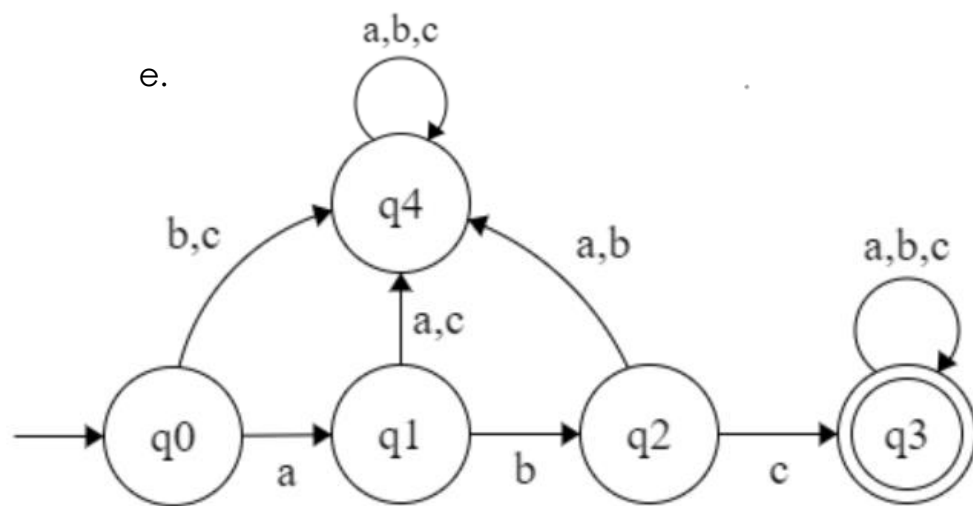
Definición Formal:



- Estados: {q0, q1 }
- Alfabeto: {0, 1}
- Función transición:

	0	1
q0	q1	q0
q1	q1	q0

- Estado inicial: {q0} No es conjunto
- Estado final: {q1}



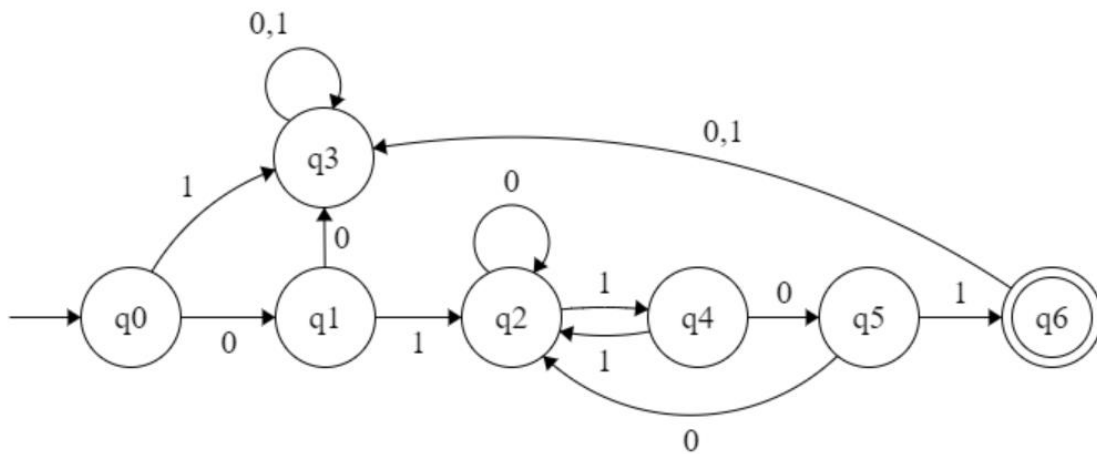
Definición Formal:

- Estados: $\{q_0, q_1, q_2, q_3, q_4\}$
- Alfabeto: $\{a, b, c\}$
- Función transición:

	a	b	c
q0	q1	q4	q4
q1	q4	q2	q4
q2	q4	q4	q3
q3	q3	q3	q3
q4	q4	q4	q4

- Estado inicial: $\{q_0\}$ no es conjunto
- Estado final: $\{q_3\}$

f. No reconoce 0101



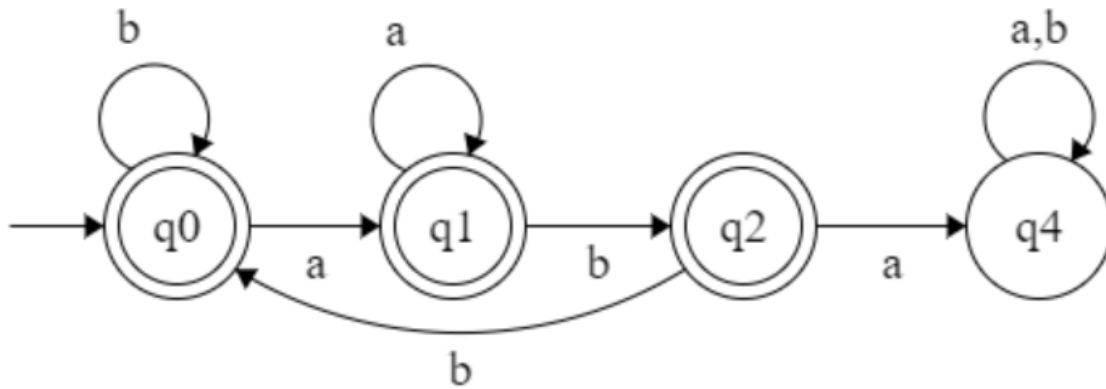
Definición Formal:

- Estados: $\{q0, q1, q2, q3, q4, q5, q6\}$
- Alfabeto: $\{0,1\}$
- Función transición:

	0	1
q0	q1	q3
q1	q3	q2
q2	q2	q4
q3	q3	q3
q4	q5	q2
q5	q2	q6
q6	q3	q3

- Estado inicial: $\{q0\}$ No es conjunto
- Estado final: $\{q6\}$

g.



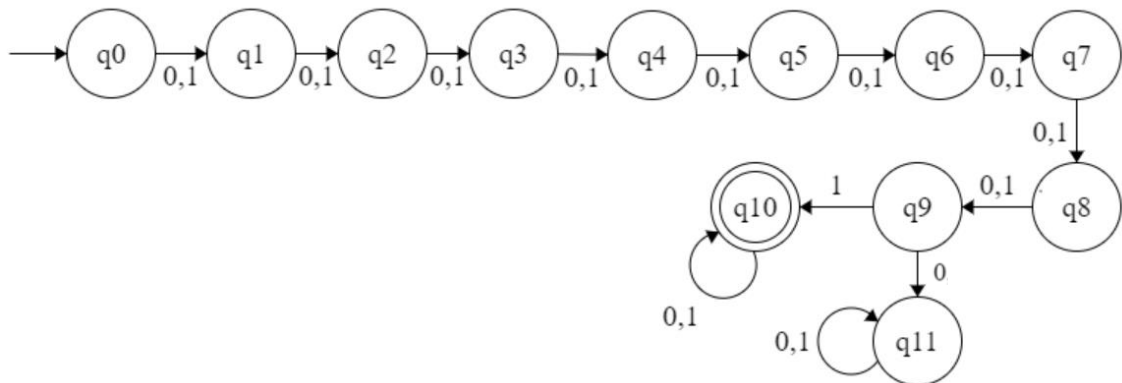
Definición Formal:

- Estados: $\{q_0, q_1, q_2, q_4\}$
- Alfabeto: $\{a, b\}$
- Función transición:

	a	b
q0	q1	q0
q1	q1	q2
q2	q4	q0
q4	q4	q4

- Estado inicial: $\{q_0\}$ No es conjunto
- Estado final: $\{q_0, q_1, q_2\}$

h.



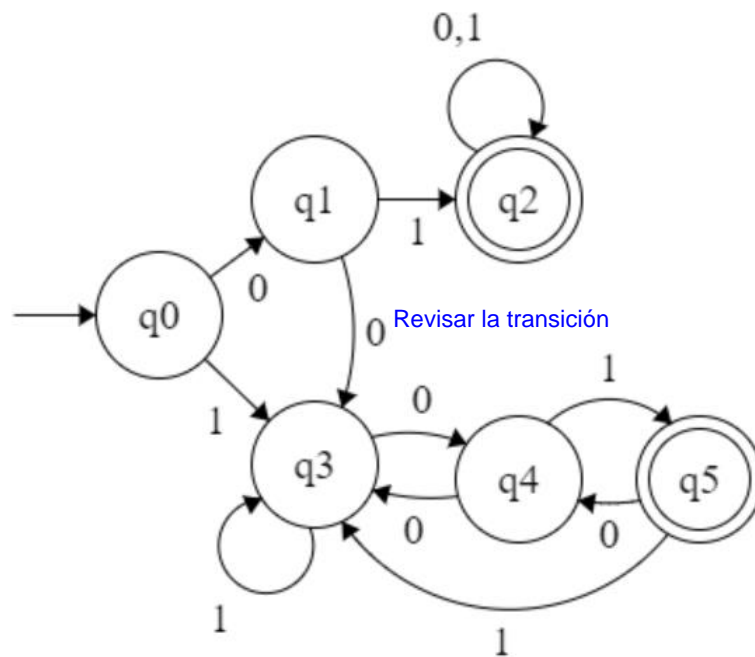
Definición Formal:

- Estados: $\{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}, q_{11}\}$
- Alfabeto: $\{0,1\}$
- Función transición:

	0	1
q0	q1	q1
q1	q2	q2
q2	q3	q3
q3	q4	q4
q4	q5	q5
q5	q6	q6
q6	q7	q7
q7	q8	q8
q8	q9	q9
q9	q11	q10
q10	q10	q10
q11	q11	q11

- Estado inicial: $\{q_0\}$ No es conjunto
- Estado final: $\{q_{10}\}$

- i. 001 no es reconocida, por ejemplo.



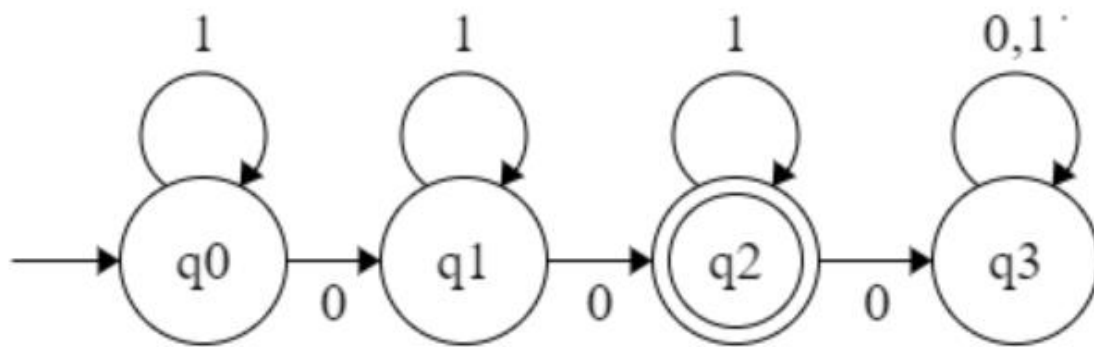
Definición Formal:

- Estados: $\{q_0, q_1, q_2, q_3, q_4, q_5\}$
- Alfabeto: $\{0, 1\}$
- Función transición:

	0	1
q0	q1	q3
q1	q3	q2
q2	q2	q2
q3	q4	q3
q4	q3	q5
q5	q4	q3

- Estado inicial: $\{q_0\}$ No es conjunto
- Estado final: $\{q_2, q_5\}$

- j. El autómata reconoce otro lenguaje, aquel cuyas cadenas contienen como máximo 2 ceros, no es el que se pide.



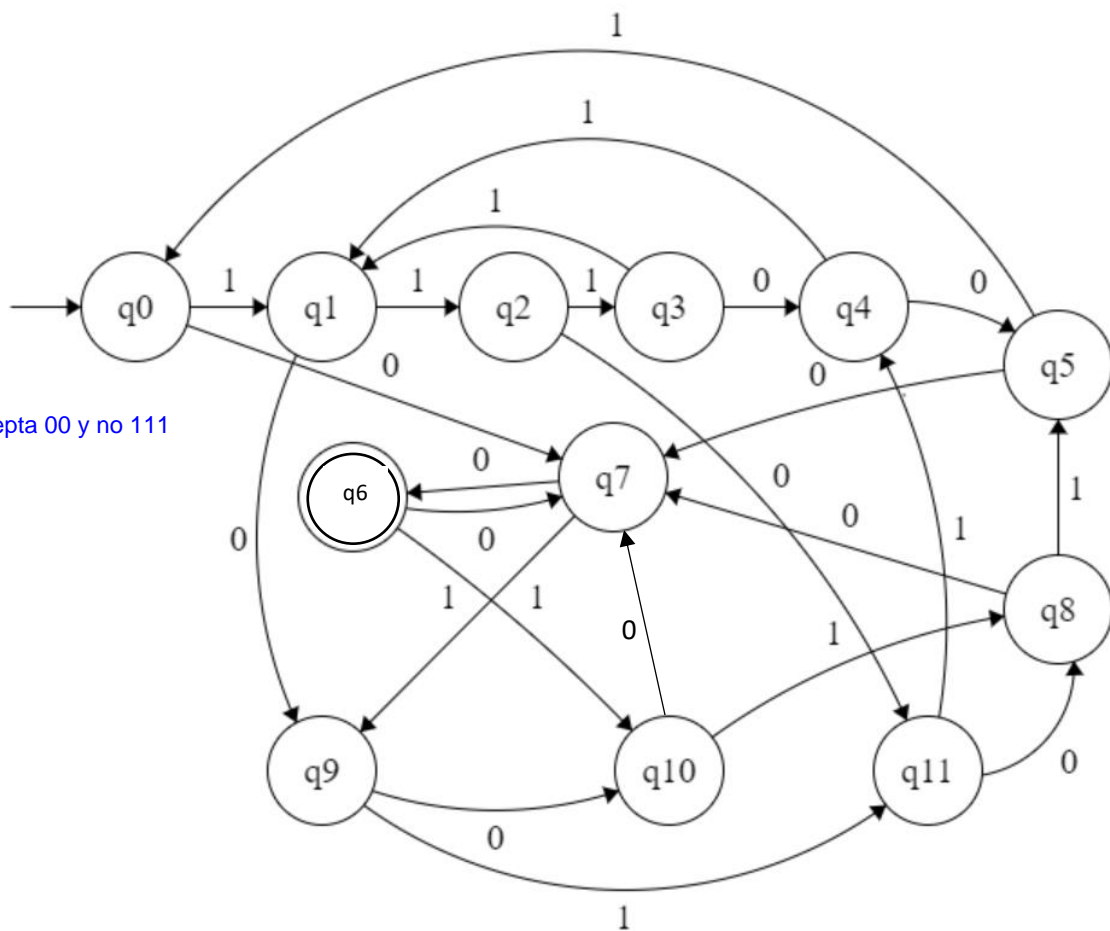
Definición Formal:

- Estados: $\{q0, q1, q2, q3\}$
- Alfabeto: $\{0, 1\}$
- Función transición:

	0	1
q0	q1	q0
q1	q2	q1
q2	q3	q2
q3	q3	q3

- Estado inicial: $\{q0\}$ No es conjunto
- Estado final: $\{q2\}$

k.



- Por qué acepta 00 y no 111
- 11100? etc.

Definición Formal:

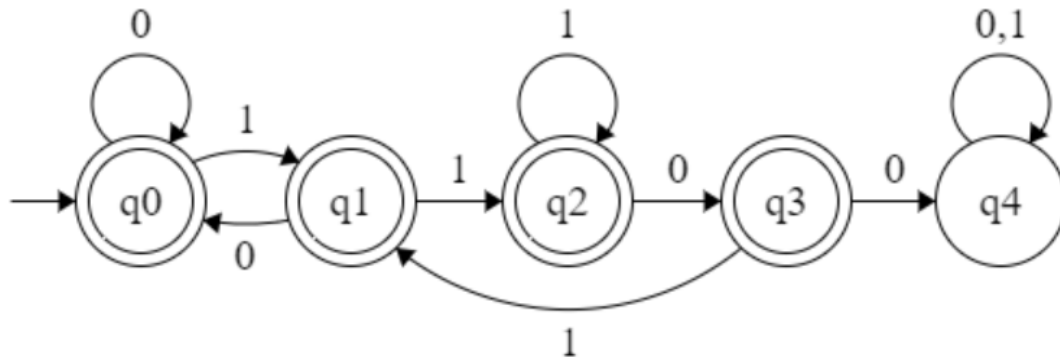
- Estados: $\{q_0, q_1, q_2, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}, q_{11}\}$
- Alfabeto: $\{0, 1\}$
- Función transición:

	0	1
q0	q7	q1
q1	q9	q2
q2	q11	q3
q3	q4	q1
q4	q5	q1
q5	q7	q0
q6	q7	q10
q7	q6	q9
q8	q7	q5
q9	q10	q11
q10	q7	q8
q11	q8	q4

- Estado inicial: $\{q_0\}$ NO es conjunto
- Estado final: $\{q_6\}$

l.

Reconoce cadenas no válidas como 110100, etc



Definición Formal:

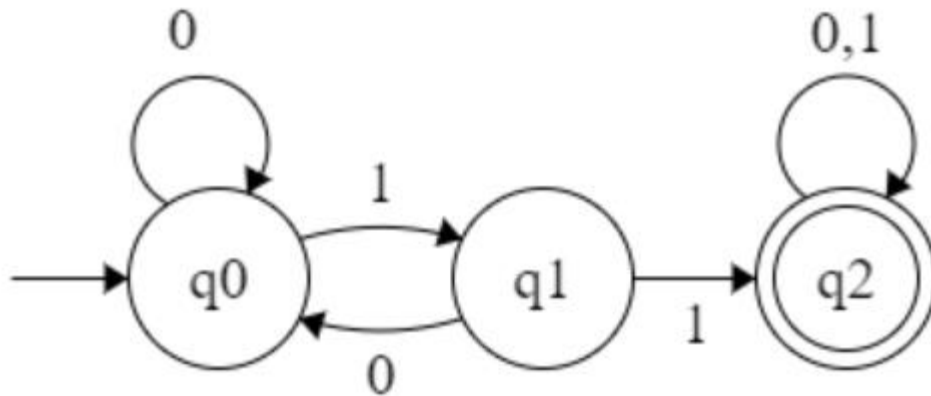
- Estados: {q0, q1, q2, q3, q4}
- Alfabeto: {0, 1 }
- Función transición:

	0	1
q0	q0	q1
q1	q0	q2
q2	q3	q2
q3	q4	q1
q4	q4	q4

- Estado inicial: {q0} No es conjunto
- Estado final: {q0, q1, q2, q3}

m.

puede contener cualquier cantidad de pares de unos consecutivos. No confundir "a lo sumo" con "como mínimo"



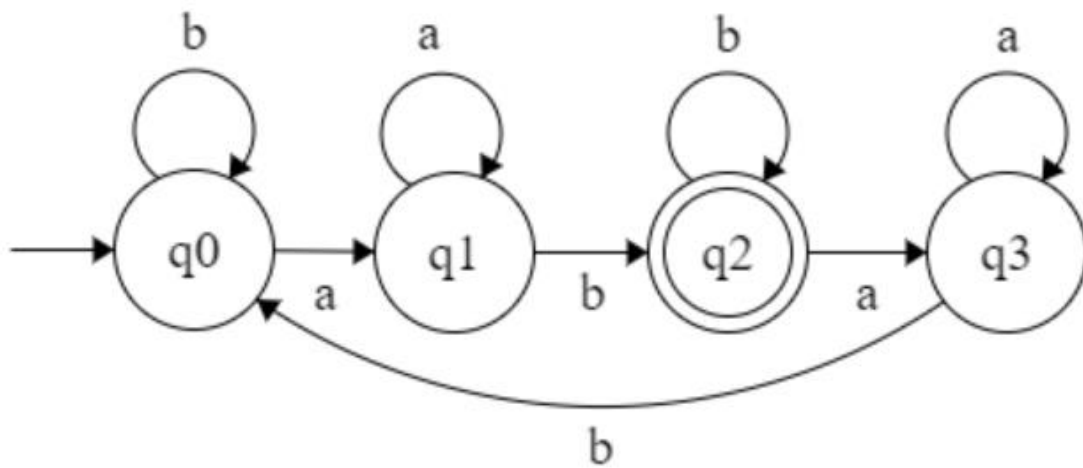
Definición Formal:

- Estados: $\{q0, q1, q2\}$
- Alfabeto: $\{0, 1\}$
- Función transición:

	0	1
q0	q0	q1
q1	q0	q2
q2	q2	q2

- Estado inicial: $\{q0\}$ No es conjunto
- Estado final: $\{q2\}$

n.



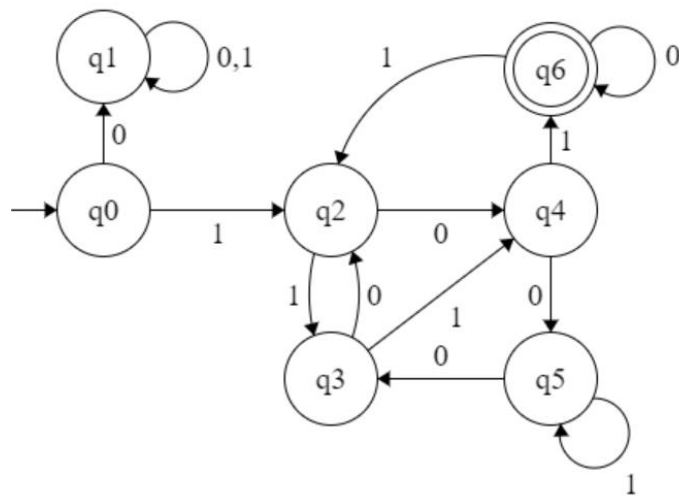
Definición Formal:

- Estados: $\{q_0, q_1, q_2, q_3\}$
- Alfabeto: $\{a, b\}$
- Función transición:

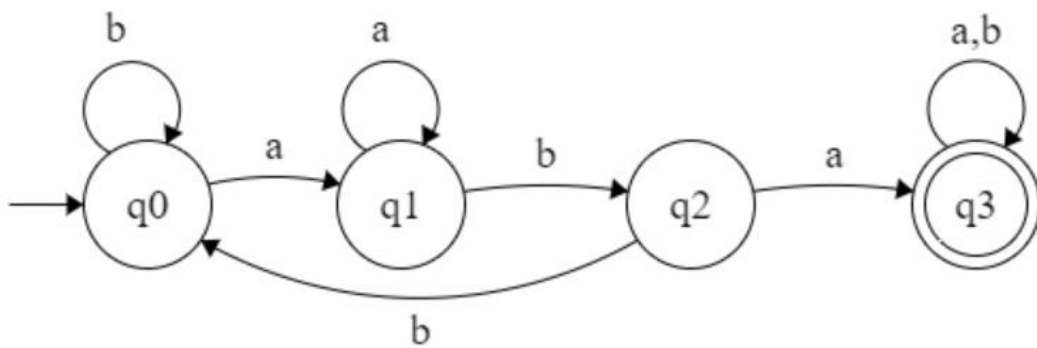
	a	b
q0	q1	q0
q1	q1	q2
q2	q3	q2
q3	q3	q0

- Estado inicial: $\{q_0\}$ No es conjunto
- Estado final: $\{q_2\}$

6.



7.



8.

Diagrama de transición que representa una luz que se enciende por un sensor de detección y se apaga en un determinado tiempo

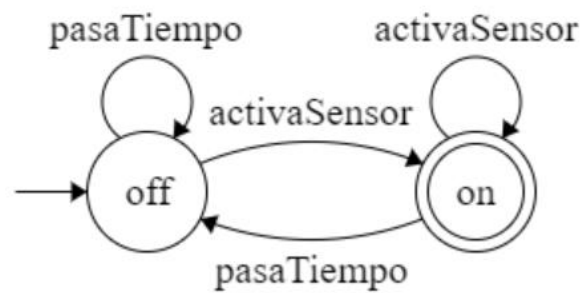


Diagrama de transición que representa el proceso de apagado de un procesador por temperatura alta.

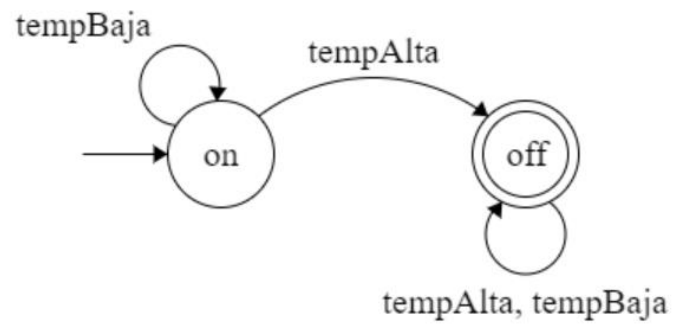
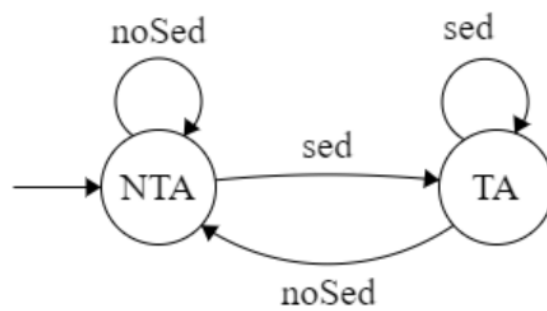
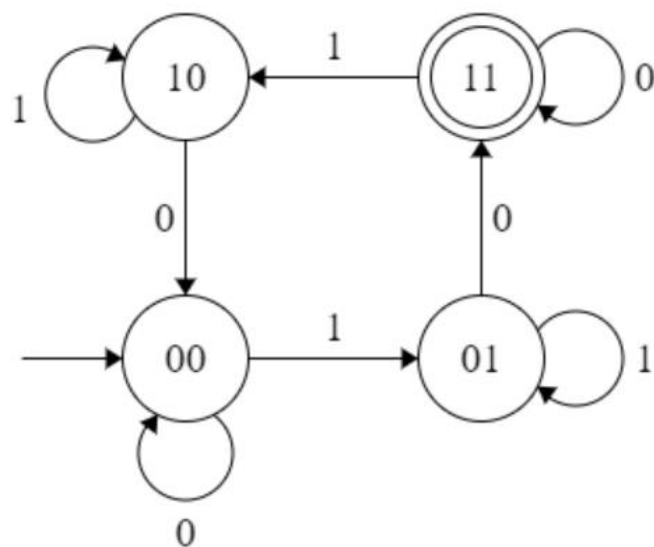


Diagrama de transición que representa el proceso de tomar agua.



9.

Parte A.



Parte B.

Esta parte del práctico se realiza en el archivo .py adjunto en la entrega.

[CODIGO FUENTE: Las cadenas no son generadas aleatoriamente por el programa.](#)