Tarea 2, Progra 2

Amado Alberto Cabrera Estrada
201905757

March 2021

Jurado

Ejercicio 3.1

Describir los lenguajes representados por las siguientes expresiones regulares definidas sobre el alfabeto $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$.

- $(\mathtt{a}|\mathtt{b})^*\mathtt{c}$
  Las cadenas aceptadas empiezan por cualquier cadena formada por `a`, `b` y ε y acaban en c.
- $(\mathtt{aa}^+)(\mathtt{bb}^*)$
  Las cadenas aceptadas empiezan por `a` de dos a infinitas veces y acaban con `b` de una a infinitas veces.
- $(\mathtt{aa}^+)|(\mathtt{bb}^*)$
  Las cadenas aceptadas son: `a` de dos a infinitas veces o `b` de una a infinitas veces (solo puede ser aceptada una u otra, no ambas a la vez).
- $\mathtt{a}^*\mathtt{b}^*\mathtt{c}^*$
  Las cadenas aceptadas llevan `a` de cero a infinitas veces, concatenado con `b` de cero a infinitas veces, concatenado con `c` de cero a infinitas veces.

Ejercicio 3.2

Representar, mediante una expresión regular, los siguientes lenguajes

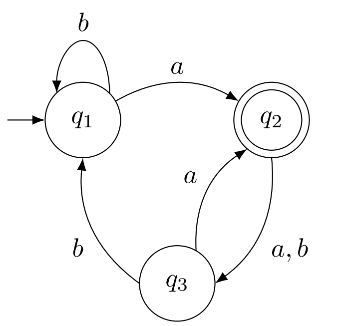
1. Considerando que $\Sigma = \{\mathtt{a}\}$
   a) el lenguaje formado por cadenas de letras `a` de longitud par
      $\mathtt{(aa)}^*$
   b) el lenguaje formado por cadenas de letras `a` de longitud impar
      $\mathtt{a(aa)}^*$

2. Considerando que $\Sigma = \{\texttt{a}, \texttt{b}\}$, el lenguaje formado por cadenas de letras a y letras b, de longitud impar, en las que se van alternando los dos símbolos, es decir, nunca aparece el mismo símbolo dos veces seguidas. Por ejemplo: `abababa` o `bab`.
`(b(ab)*)|(a(ba)*)`

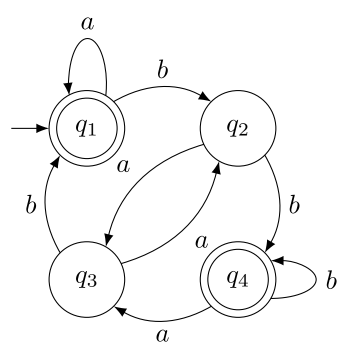
Sipser

Ejercicio 1.1

Teniendo los diagramas de estado de dos AFD, M_1 y M_2. Responda las siguientes preguntas acerca de estas máquinas.



M_1



M_2

a. ¿Cuál es el estado inicial?
   $\boldsymbol{M_1}$: El estado inicial es q_1.
   $\boldsymbol{M_2}$: El estado inicial es q_1.

b. ¿Cuál es el conjunto de estados de aceptación
   $\boldsymbol{M_1}$: El conjunto de estados de aceptación es $\{q_2\}$
   $\boldsymbol{M_2}$: El conjunto de estados de aceptación es $\{q_1, q_4\}$

c. ¿Cuál es la secuencia por los que pasa la máquina a través de la cadena `aabb`?
   $\boldsymbol{M_1}$: La secuencia sería $q_1 \to q_2 \to q_3 \to q_1 \to q_1$
   $\boldsymbol{M_2}$: La secuencia sería $q_1 \to q_1 \to q_1 \to q_2 \to q_4$

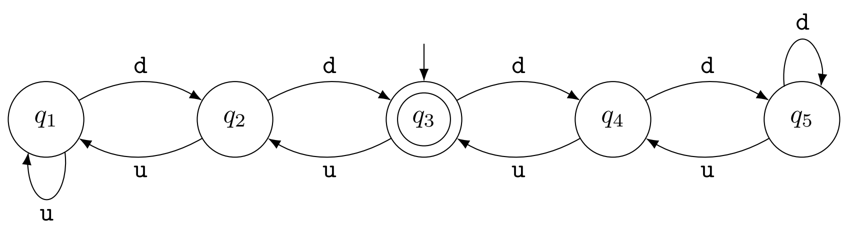
d. ¿Acepta la máquina la cadena `aabb`?
   $\boldsymbol{M_1}$: No, queda en el estado q_1 y no es de aceptación.
   $\boldsymbol{M_2}$: Sí, pues acaba en el estado q_4.

e. ¿Acepta la máquina la cadena $\boldsymbol{\varepsilon}$?
   $\boldsymbol{M_1}$: No, pues q_1 no es de aceptación.
   $\boldsymbol{M_2}$: Sí, q_1 es de aceptación.

Ejercicio 1.3

La descripción formal de un AFD M es $(\{q_1,\, q_2, q_3,\, q_4,\, q_5\},\ \{u,\, d\},\ \delta,\ q_3,\ \{q_3\})$, donde δ está dado por la siguiente tabla. Dé el diagrama de estado de esta máquina.

	u	d
q_1	q_1	q_2
q_2	q_1	q_3
q_3	q_2	q_4
q_4	q_3	q_5
q_5	q_4	q_5



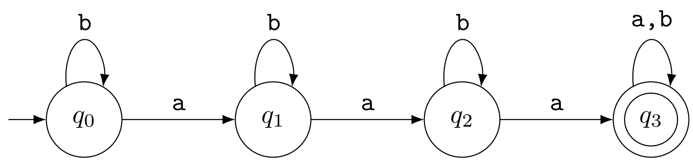
parece un elevador

Ejercicio 1.4

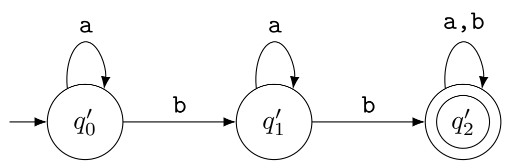
Cada uno de los siguientes lenguajes es la intersección de dos lenguajes más simples. En cada parte construya los AFD para los lenguajes simples y luego combínelos usando la construcción discutida en la nota al pie de página 3 de la página 46 para dar el diagrama de estado del AFD para el lenguaje dado. Para cada uno, $\Sigma = \{\texttt{a}, \texttt{b}\}$

a. $\{\omega | \omega$ tiene al menos tres letras `a` y al menos dos letras `b`$\}$

- $\{\omega | \omega$ tiene al menos tres letras `a`$\}$



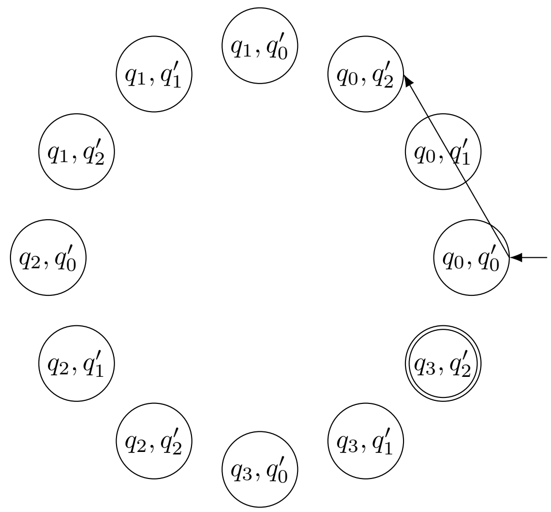
- $\{\omega | \omega$ tiene al menos dos letras `b`$\}$



- Haciendo la tabla

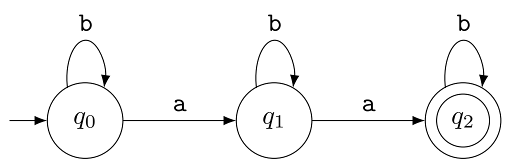
q'_0	(q_0, q'_0)	(q_1, q'_0)	(q_2, q'_0)	(q_3, q'_0)
q'_1	(q_0, q'_1)	(q_1, q'_1)	(q_2, q'_1)	(q_3, q'_1)
q'_2	(q_0, q'_2)	(q_1, q'_2)	(q_2, q'_2)	(q_3, q'_2)
$A \times B$	q_0	q_1	q_2	q_3

- Haciendo el autómata

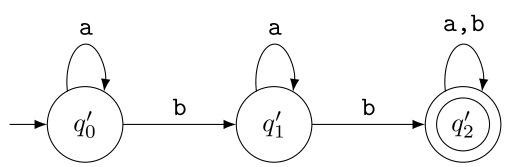


b. $\{\omega | \omega$ tiene exactamente dos letras `a` y al menos dos `b`$\}$

- $\{\omega | \omega$ tiene exactamente dos letras `a`$\}$



- $\{\omega | \omega$ tiene al menos dos `b`$\}$



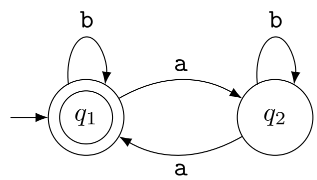
- Haciendo la tabla

q_0'	(q_0, q_0')	(q_1, q_0')	(q_2, q_0')
q_1'	(q_0, q_1')	(q_1, q_1')	(q_2, q_1')
q_2'	(q_0, q_2')	(q_1, q_2')	(q_2, q_2')
$A \times B$	q_0	q_1	q_2

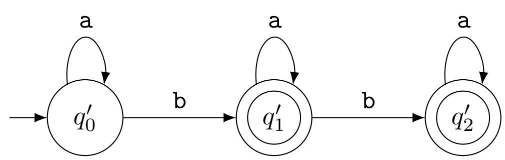
- Haciendo el autómata

c. $\{\omega | \omega$ tiene un número par de letras `a` y una o dos letras `b`$\}$

- $\{\omega | \omega$ tiene un número par de letras `a`$\}$



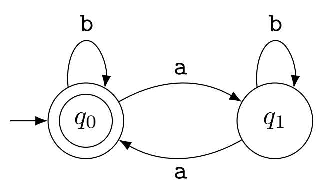
- $\{\omega|\omega$ tiene una o dos letras `b`$\}$



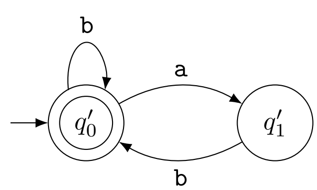
- Haciendo la tabla
- Haciendo el autómata

d. $\{\omega|\omega$ tiene un número par de letras `a` y cada `a` se sigue por al menos de una `b`$\}$

- $\{\omega|\omega$ tiene un número par de letras `a`$\}$



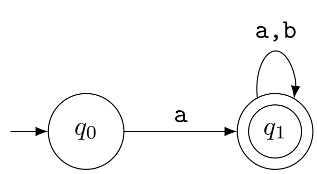
- $\{\omega|\omega$ cada `a` se sigue por al menos de una `b`$\}$



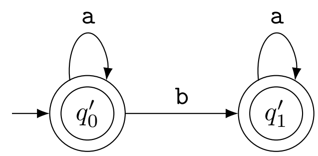
- Haciendo la tabla
- Haciendo el autómata

e. $\{\omega|\omega$ comienza con una `a` y tiene como máximo una `b`$\}$

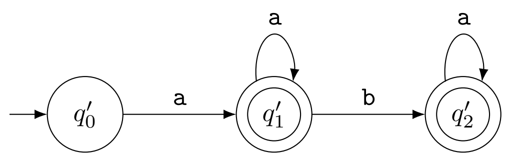
- $\{\omega|\omega$ comienza con una `a`$\}$



- $\{\omega | \omega$ tiene como máximo una b$\}$

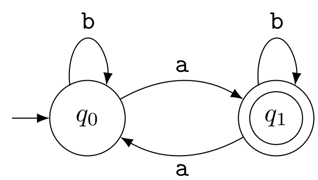


- Haciendo la tabla
- Haciendo el autómata

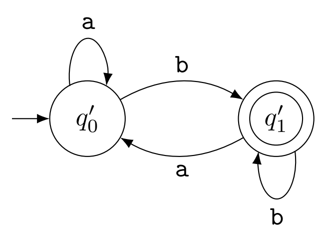


f. $\{\omega | \omega$ tiene un número impar de letras a y termina con una b$\}$

- $\{\omega | \omega$ tiene un número impar de letras a$\}$



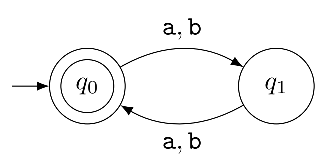
- $\{\omega | \omega$ termina con una b$\}$



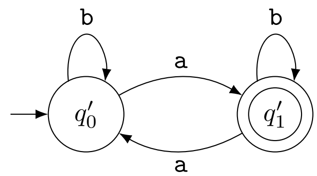
- Haciendo la tabla
- Haciendo el autómata

g. $\{\omega | \omega$ tiene longitud par y un número impar de letras a$\}$

- $\{\omega | \omega$ tiene longitud par$\}$



- $\{\omega|\omega$ tiene un número impar de letras a$\}$

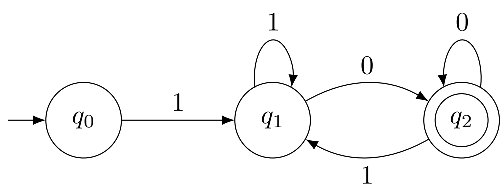


- Haciendo la tabla
- Haciendo el autómata

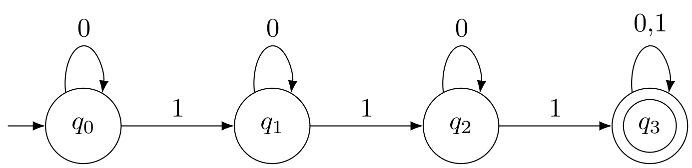
Ejercicio 1.6

Dar los diagramas de estado de los AFD que reconocen los siguientes lenguajes. Para cada uno el alfabeto es $\{0, 1\}$

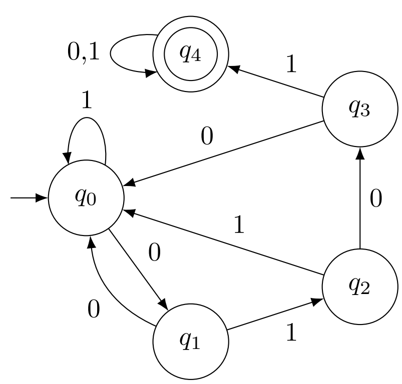
a. $\{\omega|\omega$ empieza con 1 y acaba con 0$\}$



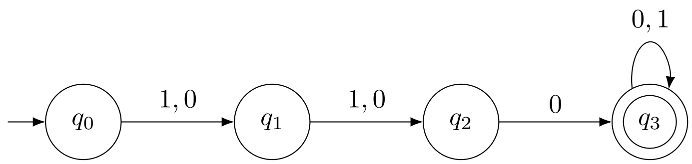
b. $\{\omega|\omega$ contiene al menos tres números 1$\}$



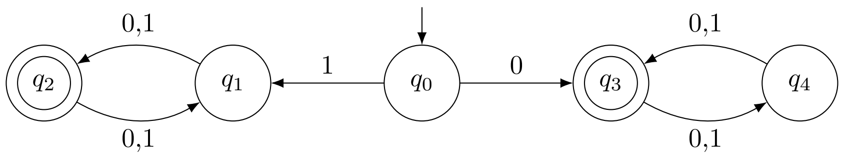
c. $\{\omega|\omega$ contiene la subcadena 0101$\}$



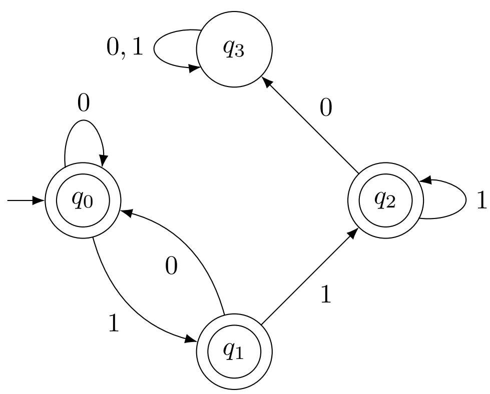
d. $\{\omega|\omega$ tiene longitud mayor o igual a 3 y su tercer símbolo es $0\}$



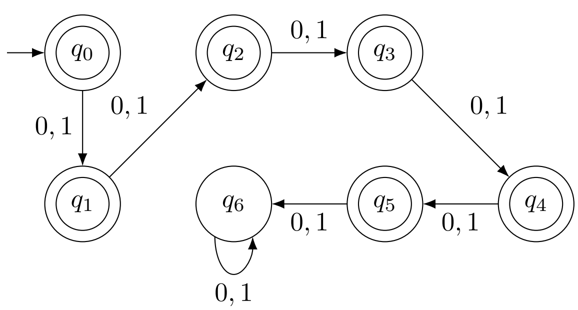
e. $\{\omega|\omega$ empieza con 0 y tiene longitud impar, o empieza con uno y tiene longitud par$\}$



f. $\{\omega|\omega$ no contiene la subcadena $110\}$

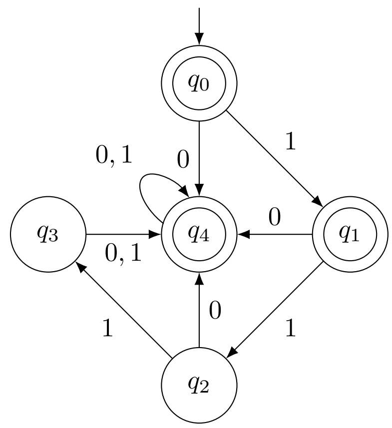


g. $\{\omega|$la longitud de ω es a lo mucho $5\}$

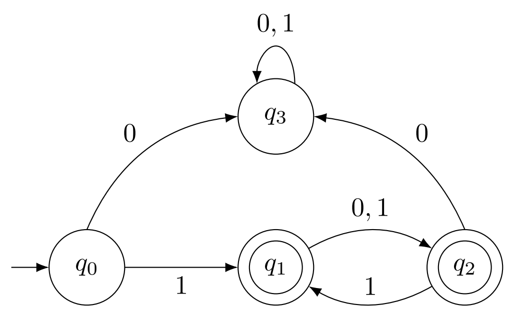


Parece una tortuguita.

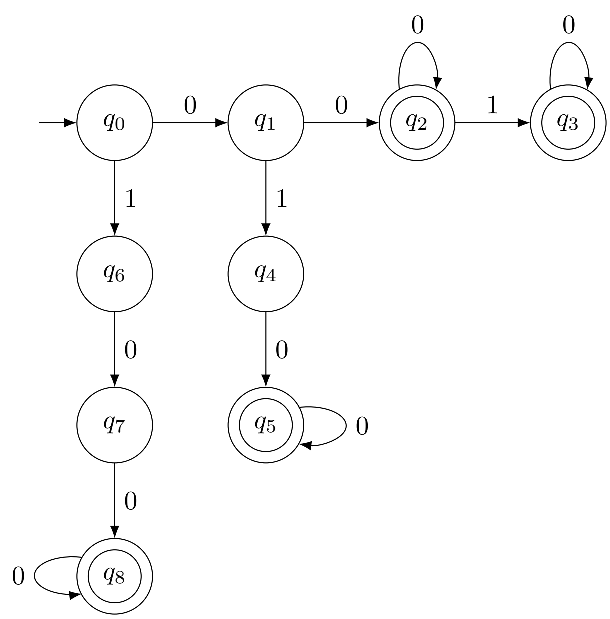
h. $\{\omega|\omega$ es cualquier cadena excepto 11 y $111\}$



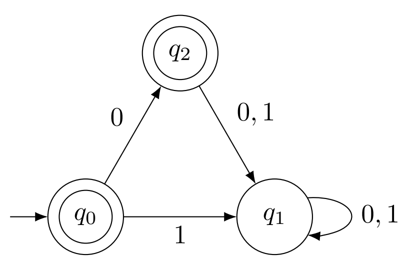
i. $\{\omega|$cualquier posición impar de ω es un $1\}$



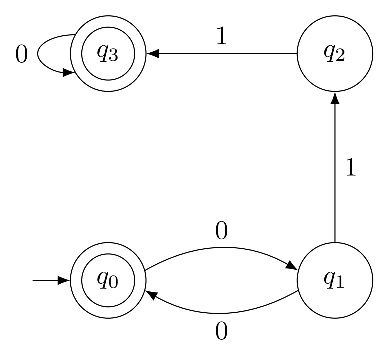
j. $\{\omega|\omega$ contiene al menos dos números 0 y como mucho un $1\}$



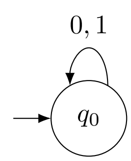
k. $\{\varepsilon, 0\}$



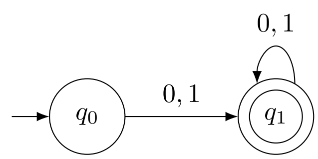
l. $\{\omega|\omega$ contiene un número par de números 0, o contiene exactamente dos números 1$\}$



m. El conjunto vacío.



n. Todas las cadenas menos la cadena vacía



Ejercicio 1.8

Use la construcción en la demostración del teorema 1.45 para dar los diagramas de estado de los AFND reconociendo la unión de los lenguajes descritos en

a. Ejercicios 1.6a y 1.6b.

b. Ejercicios 1.6c y 1.6f.

Ejercicio 1.11

Demuestre que todo AFND puede ser convertido a uno equivalente que tenga un solo estado de aceptación.

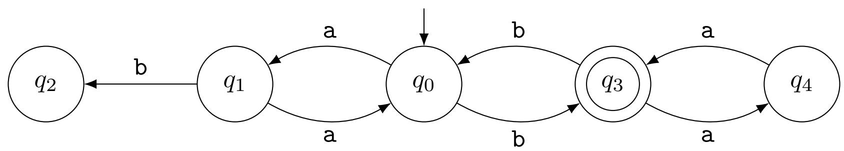
Teniendo un AFND $N = \{Q, \Sigma, \delta, s, F\}$ tal que la cardinalidad de F es mayor a 1. Construiremos un nuevo AFND N' tal que $N' = \{Q', \Sigma', \delta', s', F'\}$ en donde:

- $Q' = Q \cup \{q_a\}$
- $\Sigma = \Sigma'$
- $\delta'(q, r) = \begin{cases} \delta(q, r) & q \notin F \text{ y } r \neq \epsilon \\ q_a & q \in F \text{ y } r = \epsilon \end{cases}$
- $s' = s$
- $F' = \{q_a\}$

Esto quiere decir que se conecta cada estado aceptado por N con un nuevo estado único de aceptación $\{q_a\}$ y se puede llegar a él a partir de una ε–transición, por lo tanto se unifican todos los estados de aceptación en uno solo. Similar a lo que hacemos para transformar un autómata a una expresión regular.

Ejercicio 1.12

Sea $D = \{w|w$ contiene un número par de letras `a`, un número impar de letras `b` y no contiene la subcadena `ab`$\}$. Dé un AFD con cinco estados que reconozca D y una expresión regular que genere a D. (Sugerencia: Describa D de manera más simple).



Ejercicio 1.14

a) Demuestre que si M es un AFD que reconoce el lenguaje B, intercambiando el estado de aceptación y de no aceptación en M produce un nuevo AFD reconociendo el complemento de B. Concluya que la clase de lenguajes regulares es cerrado bajo complemento.

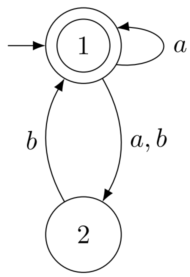
   Teniendo un AFD $M = \{Q, \Sigma, \delta, s, F\}$ tal que $L(M) = B$ entonces intercambiar los estados de aceptación por no aceptación y viceversa significa que todas las cadenas no aceptadas ahora lo serán, formando el lenguaje A. Tal que $\Sigma^* = A \cup B$ y $A \cap B = \varnothing$, esto significa que $A = B^c$. Lo que nos daría un nuevo autómata $M' = \{Q', \Sigma', \delta', s', F'\}$ tal que:

   - $Q' = Q$
   - $\Sigma' = \Sigma$
   - $\delta' = \delta$
   - $s' = s$
   - $F' = Q \backslash F$

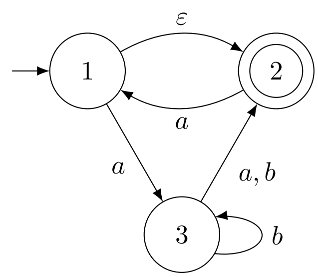
b) Demuestre dando un ejemplo de que si M es un AFND que reconoce al lenguaje C, intercambiando los estados de aceptación y no aceptación en M no necesariamente produce un nuevo AFND que reconozca el complemento de C. ¿Es la clase de lenguajes reconocidos por por ANFD cerrado bajo complemento? Explique su respuesta.

Ejercicio 1.16

Use la construcción dada en el teorema 1.39 para convertir el siguiente autómata finito no determinista a su equivalente autómata finito determinista.



(a)



(b)

Ejercicio 1.17

a. Dé un ANFD reconociendo el lenguaje $(\mathtt{01}\cup\mathtt{001}\cup\mathtt{010})^*$.

b. Convierta este ANFD a un AFD equivalente. Dé únicamente la porción del AFD que es alcanzable desde el estado inicial.

Ejercicio 1.19

Use el procedimiento descrito en el Lema 1.55 para convertir la siguiente expresión regular a un autómata finito no determinista.

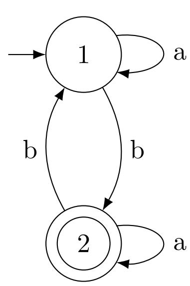
a. $(\mathtt{0}\cup\mathtt{1})^*\mathtt{000}(\mathtt{0}\cup\mathtt{1})^*$

b. $(((\mathtt{00})^*(\mathtt{11}))\cup\mathtt{01})^*$

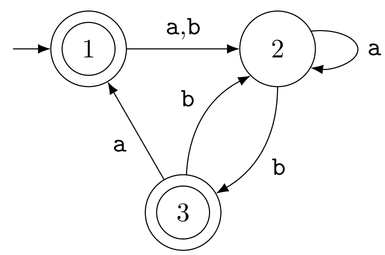
c. $\emptyset^*$

Ejercicio 1.21

Use el procedimiento descrito en el Lema 1.60 para convertir el siguiente autómata finito a expresiones regulares.



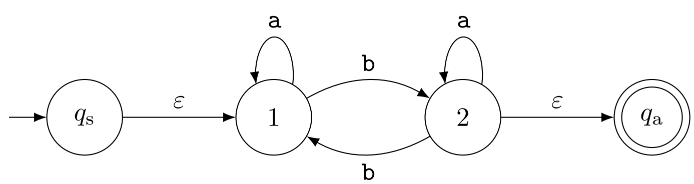
(a)



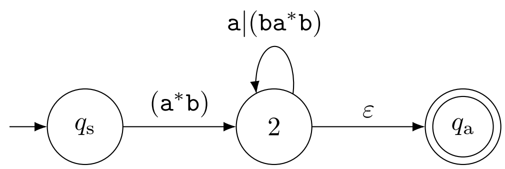
(b)

a) Primer grafo

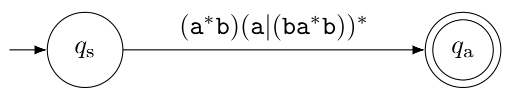
- Añadiendo el estado inicial y final.



- Tomando $q_{\text{rip}} = 1$.

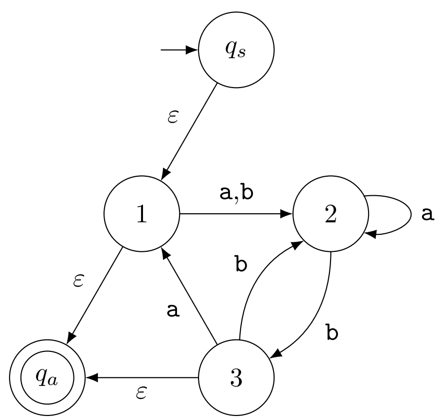


- Tomando $q_{\text{rip}} = 2$.

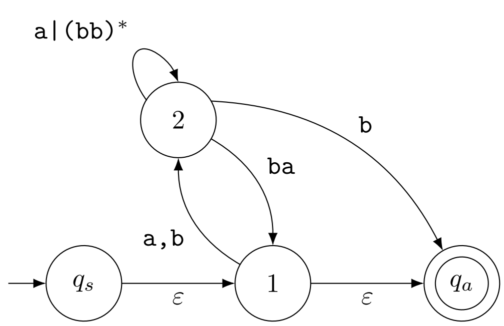


b) Segundo grafo

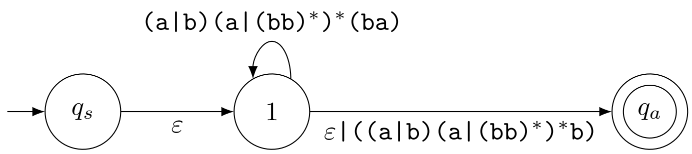
- Añadiendo estado inicial y final



- Tomando $q_{\text{rip}} = 3$.



- Tomando $q_{\text{rip}} = 2$.



- Tomando $q_{\text{rip}} = 1$.