1. Determinar si los siguientes lenguajes son regulares o libres de contexto. Justificar las respuestas.

- $\{0^i b^j \mid i = 2j \text{ ó } 2i = j\}$
- $\{uu^{-1} \mid u \in \{0,1\}^*, |u| \leq 1000\}$
- $\{uu^{-1} \mid u \in \{0,1\}^*, |u| \geq 1000\}$
- $\{0^i 1^j 2^k \mid i = j \text{ ó } j = k\}$

v es arbitrario entonces no lo puedo fijar

$\dashv$ Veamos que no es regular por el Lema de Bombeo, $n \in \mathbb{N}$, $z = 0^n b^{n/2}$ ($i = 2j$), $u = 0^{n-1}$ $v = 0$ $w = b^{n/2}$

$z = 0^{2n} b^n$, $u = 0^k$ $v = 0^\ell$ $w = 0^{2n-k-\ell} b^n$

i) $|uv| = n$ (en particular $|uv| \leq n$)

ii) $|v| = 1$ (en particular $|v| \geq 1$)

⟶ se cumplen hipótesis

tomando $i = 2 \Rightarrow uv^i w = 0^k v^{2\ell} 0^{2n-k-\ell} b^n = 0^{2n+\ell} b^n \notin \mathcal{L}$

Tomando $i = 2$ vemos que $uv^i w \notin L$ pues $uv^i w = 0^{n+1} b^{n/2}$ y $n+1 \neq n$

(2 en 1) ⇝ El primero es regular, el siguiente no.

$\dashv$ Veamos que ni este ni el siguiente son regulares con la misma idea mediante el Lema de Bombeo tomando $n \in \mathbb{N}$ $z = (0101)^{\frac{n}{4}} (1010)^{\frac{n}{4}}$ con $n \leq 500$, supongamos $n = 1$ para el primer lenguaje, basta tomar $u = (010101)^{\frac{n}{4}-1}$ $v = (0101)$ $w = (1010)^{n/4}$ entonces, es fácil ver que $uv^2 w \notin L$ pues

$uv^2 w = (0101)^{n/4} (01)^2 (1010)^{n/4}$

De la misma manera se haría para el siguiente con $n > 1000$ $z = (0101)^n (1010)^n$

$\dashv$ Veamos que este tampoco es regular por el Lema de Bombeo tomando $z = 0^n 1^n 2^n$, $n \neq n$; como sabemos que $\mathcal{L} = \{0^i 1^i \mid i \in \mathbb{N}\}$ no es regular entonces con la misma idea este tampoco es regular.

2. Determinar qué lenguajes son regulares o libres de contexto de los siguientes:

a) $\{u0u^{-1} \mid u \in \{0,1\}^*\}$

b) Números en binario que sean múltiplos de 4

c) Palabras de $\{0,1\}^*$ que no contienen la cabcadena 0110

a) No es regular, por el Lema de Bombeo, $n \in \mathbb{N}$ $z = 1^n 0 1^n \Rightarrow u = 1^{n-1}$ $v = 1$ $w = 01^n$ cumple i) y ii)

pero $i = 2$ $uv^2 w = 1^{n+1} 0 1^n \notin L$

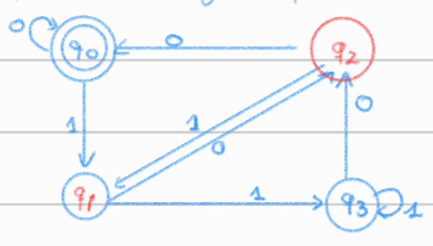
$u = 1^\ell$, $v = 1^k$, $w = 1^{n-\ell-k} 0 1^n \Rightarrow$

$\ell + k \leq n$

$k \geq 1$

$i = 2 \Rightarrow uv^2 w = 1^\ell 1^{2k} 1^{n-\ell-k} 0 1^n = 1^{n+k} 0 1^n \notin \mathcal{L}$

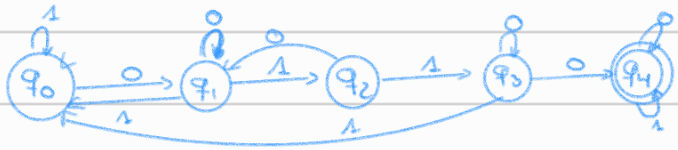
b) Veamos que sí es regular; para ello tenemos el AFD

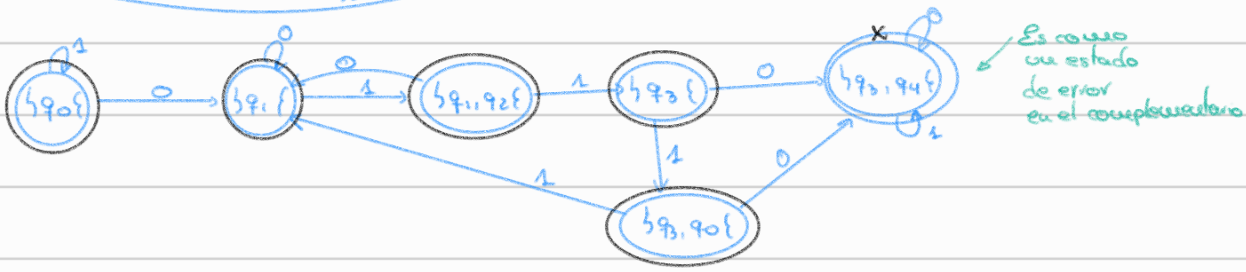


Debemos siempre acabar en 00

como $k \geq 1$

c)





Tomamos el complementario sabiendo que podemos obtener una gramática regular

$q_0 \longrightarrow 1q_0 \mid 0q_1 \mid \varepsilon$ $\qquad A = \{q_1, q_2\}$

$q_1 \longrightarrow 0q_1 \mid 1A \mid \varepsilon$ $\qquad B = \{q_0, q_3\}$

$A \longrightarrow 0q_1 \mid 1q_3 \mid \varepsilon$

$q_3 \longrightarrow 1B \mid \varepsilon$

$B \longrightarrow 1q_1 \mid \varepsilon$

3. Determinar qué lenguajes son regulares y qué lenguajes son libres de contexto entre los siguientes:
   a) Conjunto de palabras sobre el alfabeto $\{0, 1\}$ en las que cada 1 va precedido por un número par de ceros. → sólo un nº par
   b) Conjunto $\{0^i1^2j0^{i+j} | i, j \geq 0\}$
   c) Conjunto $\{0^i1^j0^{i*j} | i, j \geq 0\}$

$\boxed{\text{Sí es reg } ((00)^+1)^*0^*}$

a) $\mathcal{L} = \{(0^i\ 1)^j 0^k \in \{0,1\}^* \mid i = 2n, j, k \in \mathbb{N}\}$; veamos que no es regular, para ello, tomaremos la palabra $z = 0^n 1$ con $n = 2k, k \in \mathbb{N}$, tomando $z = uvw$ con $u = 0^{n-1}$ $v = 0$ $w = 1$ vemos que se cumple

i) $|uv| \leq n$; $|uv| = n$

ii) $|v| \geq 1$; $|v| = 1$

Pero tomando $i = 2$ obtenemos que $uv^2w = 0^{n+1}1$ luego como n es par ⇒ n+1 es impar

b) $\{0^i 1^{2j} 0^{i+j} \mid i, j \geq 0\}$ veamos que tampoco es regular tomando $z = 0^n 1^{2m} 0^{n+m}$ tomando $u = 0^k$ $v = 0^t$ | $k + t \leq n$ obtenemos que i) y ii) se cumplen; entonces veamos que $\exists i \geq 0 \mid uv^iw \notin \mathcal{L}$

$w = 0^{n-k-t} 1^{2m} 0^{n+m}$ $\quad k \geq 1$

Basta tomar $i = 2$, es decir, $uv^2w = 0^k 0^{2t} 0^{n-k-t} 1^{2m} 0^{n+m} \notin \mathcal{L}$

$= 0^{n+t} 1^{2m} 0^{n+m}$

c) Tampoco es regular y basta tomar $z = 0^n 1^n 0^{n^2}$ y tomar $u = 0^k$ $v = 0^t$ | $k + t \leq n$ (i) (ii)

$w = 0^{n-k-t} 1^n 0^{n^2}$

Veamos ahora que $\exists i \geq 0$ tal que $uv^iw \notin \mathcal{L}$; $i = 2$ $uv^iw = 0^k 0^{2t} 0^{n-k-t} 1^n 0^{n^2} = 0^{n+t} 1^n 0^{n^2}$

y $(n+t)n \neq n^2$

4. Determina si los siguientes lenguajes son regulares. Encuentra una gramática que los genere o un reconocedor que los acepte.

   a) $L_1 = \{0^i 1^j : j < i\}$.

   b) $L_2 = \{001^i 0^j 11 : i, j \geq 1\}$.

   c) $L_3 = \{010u : u \in \{0,1\}^*, u$ no contiene la subcadena 010$\}$.

a) No es regular; tomando $z = 0^{n+1} 1^n$ $n \geq 0$

$u = 0^k$ $v = 0^t$ $1 \leq k+t \leq n$; $w = 0^{n+1-k-t} 1^n$

basta tomar $i=0$ para ver que $uv^0w = 0^k 0^{n+1-k-t} 1^n = 0^{n+1-t} 1^n$ y $n+1-t \leq n$ ya que $t \geq 1$

b) $S \longrightarrow 001A$ $\quad 00\,1^t 0^r 11$

$A \longrightarrow 1A \mid 0B$

$B \longrightarrow 0B \mid 11$

c) $S \longrightarrow 010A$ $\quad$ La palabra que no contiene 010 es u

$A \longrightarrow \varepsilon \mid 1A \mid 0B$

$B \longrightarrow \varepsilon \mid 0B \mid 1C$

$C \longrightarrow 1A \mid \varepsilon$

5. Sea el alfabeto $A = \{0, 1, +, =\}$, demostrar que el lenguaje

$$ADD = \{x = y + z \mid x, y, z \text{ son números en binario, y } x \text{ es la suma de } y \text{ y } z\}$$

no es regular.

Veamos que no es regular haciendo uso del lema de Bombeo; para ello tomamos $n>0$ tal que $z =$ "$1^n = 0^n + 1^n$" es una palabra de ADD

Tomando $u = 1^k$ $v = 1^t$ tal que $k+t \leq n$ queda $w =$ "$1^{n-k-t} = 0^n + 1^n$" sabemos que se cumple

i) $|uv| = k+t \leq n$

ii) $|v| = t \geq 1$

Pero tomando $i=2$ vemos que $uv^2w =$ "$1^{n+t} = 0^n + 1^n$" $\notin ADD$ pues $1^{n+t} \neq 1^n$

6. Determinar si los siguientes lenguajes son regulares o no:

   a) $L = \{uvu^{-1}$ tal que $u, v \in \{0,1\}^*\}$. → restricción sobre $v \neq \varepsilon$

   b) L es el lenguaje sobre el alfabeto $\{0,1\}$ formado de las palabras de la forma $u0v$ donde u^{-1} es un prefijo de v.

   c) L es el lenguaje sobre el alfabeto $\{0,1\}$ formado por las palabras en las que el tercer símbolo empezando por el final es un 1.

a) Tampoco es regular, misma demostración que uu^{-1} 3

b) $L = \{u0u^{-1}v \in \{0,1\}^*\}$

Usando el apartado a) vemos que no es regular 3

c) $S \longrightarrow 0A \mid 1A$ 3

$A \longrightarrow S \mid 1B$

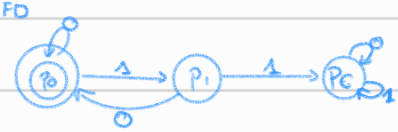
$B \longrightarrow 0C \mid 1C$

$C \longrightarrow 0 \mid 1$

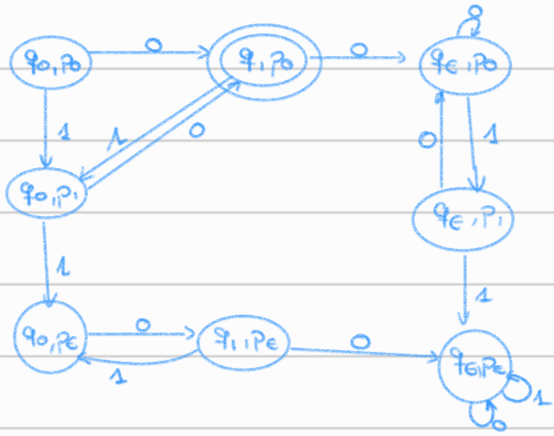
8. Dar una expresión regular para la intersección de los lenguajes asociados a las expresiones regulares $(\mathbf{01}+\mathbf{1})^*\mathbf{0}$ y $(\mathbf{10}+\mathbf{0})^*$. Se valorará que se construya el autómata que acepta la intersección de estos lenguajes, se minimice y, a partir del resultado, se construya la expresión regular.

$(01+1)^*0$: AFD

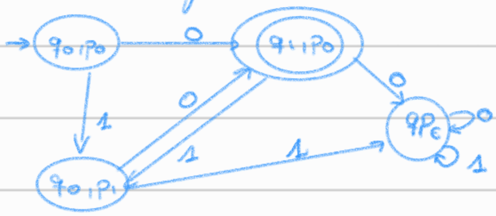
$(10+0)^*$ AFD



Intersección



Obtenemos el minimal juntando todos los estados de error



3/

$\{q_0,p_0\} = a$; $\{q_0,p_1\} = b$, $\{q_1,p_0\} = c$

$$\begin{cases} a = 0c + 1b \longrightarrow a = 0(10)^* + 10(10)^* = (\varepsilon + 1)0(10)^* \\ b = 0c \longrightarrow b = 0(10)^* \\ c = \varepsilon + 1b \longrightarrow c = \varepsilon + 10c \overset{\text{Arden}}{\Longrightarrow} c = (10)^* \end{cases}$$

$= 0(10)^* + (10)^+$

10. Encontrar un AFD minimal para el lenguaje

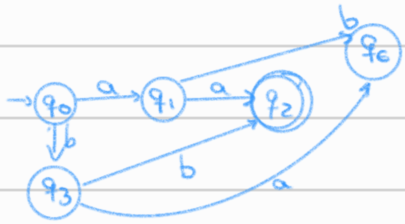
$$(a+b)^*(aa+bb)(a+b)^*$$

→ Mejor construir el autómata de golpe, se piensa mejor así

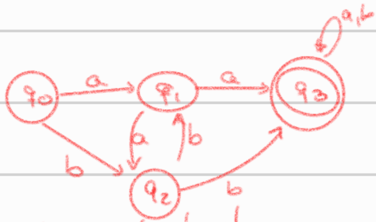
Para ello construimos el AFD de cada expresión regular por separado

$\boxed{(a+b)^*}$

$\boxed{(aa+bb)}$

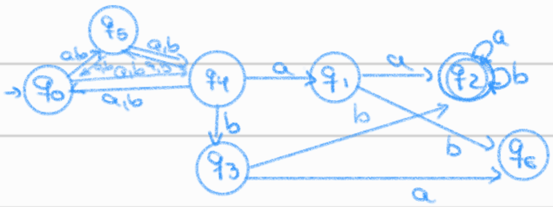


A palo.

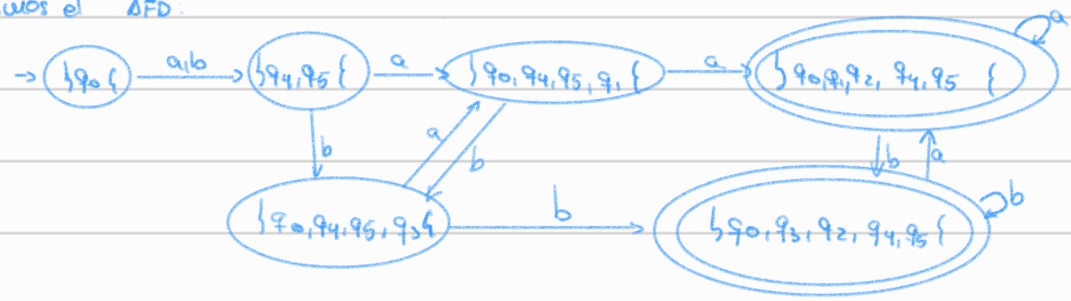


Aplicando el algoritmo

Armamos el autómata finito no determinista de la concatenación



Construimos el AFD:



No me deja hacer todas las posibilidades que quiero.

11. Para cada uno de los siguientes lenguajes regulares, encontrar el autómata minimal asociado, y a partir de dicho autómata minimal, determinar la gramática regular que genera el lenguaje:

- a^+b^+
- $a(a+b)^*b$

a) a^+:

b^+

a^+b^+

Ya hemos quitado los estados indistinguibles

gramática regular

$S \longrightarrow aA$

$A \longrightarrow aA \mid bB$

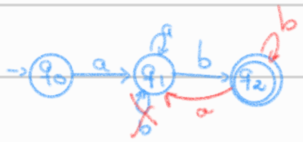
$B \longrightarrow bB \mid b$

b) a:

b:

$(a+b)^*$

Por tanto, el AFND → no es AFD



Donde la gramática es:

$S \longrightarrow aA$

$A \longrightarrow aA \mid bA \mid b$

13. Determinar autómatas minimales para los lenguajes $L(M_1) \cup L(M_2)$ y $L(M_1) \cap \overline{L(M_2)}$ donde,

- $M_1 = (\{q_0, q_1, q_2, q_3\}, \{a, b, c\}, \delta_1, q_0, \{q_2\})$ donde

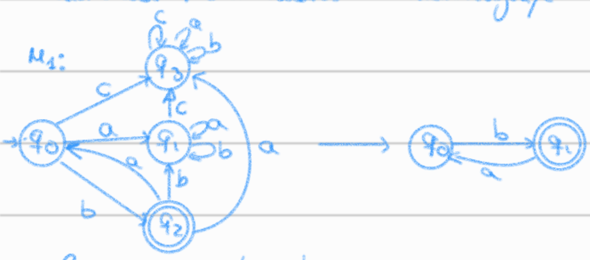
δ_1	q_0	q_1	q_2	q_3
a	q_1	q_1	q_3	q_3
b	q_2	q_1	q_1	q_3
c	q_3	q_3	q_0	q_3

- $M_2 = (\{q_0, q_1, q_2, q_3\}, \{a, b, c\}, \delta_2, q_0, \{q_2\})$

δ_2	q_0	q_1	q_2	q_3
a	q_1	q_1	q_3	q_3
b	q_1	q_2	q_2	q_3
c	q_3	q_3	q_0	q_3

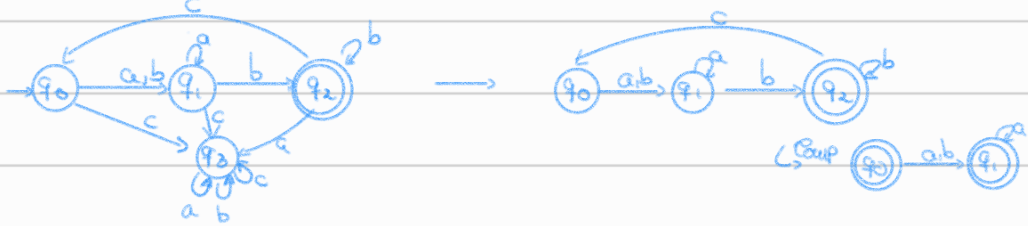
Construimos los autómatas de cada lenguaje:

M_1:



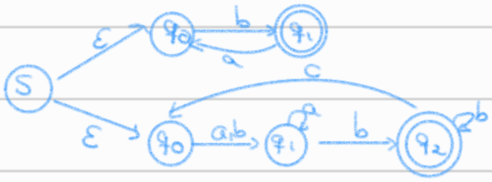
Expresión regular: $b(ab)^*$

M_2:



Expresión regular $((a+b)a^*b^+)(c((a+b)a^*b^+))^*$

Para obtener $L(M_1) \cup L(M_2)$; como εAFND. Hay que minimizar

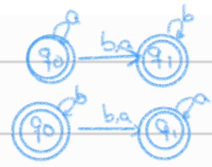


Para obtener $L(M_1) \cap \overline{L(M_2)}$: → Sacar producto ambos y minimizar

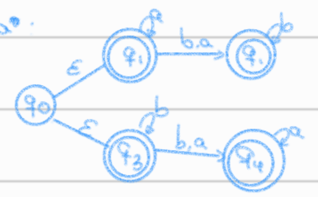
14. Dado el conjunto regular representado por la expresion regular $a^*b^* + b^*a^*$, construir un autómata finito determinístico minimal que lo acepte.

a^*b^*:

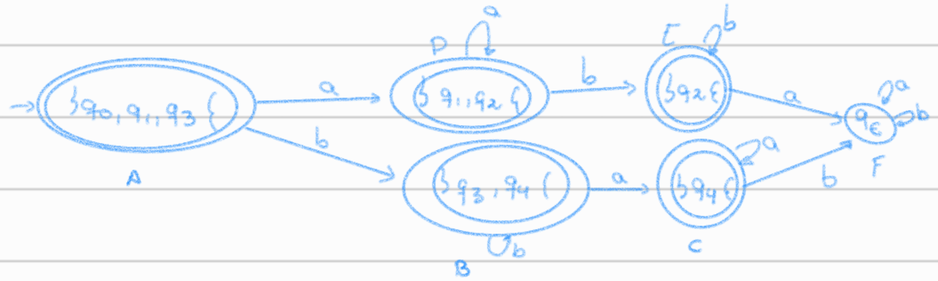
b^*a^*:



$a^*b^* + b^*a^*$:



AFD:



	A	B	C	D	E
B	X				
C	X	X			
D	X	X	X		
E	X	X	X	X	
F	X	X	X	X	X

Por tanto, es minimal

SALIR

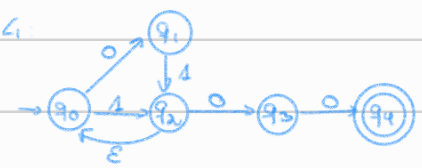
15. Sean los lenguajes:

- $L_1 = (01 + 1)^*00$
- $L_2 = 01(01 + 1)^*$

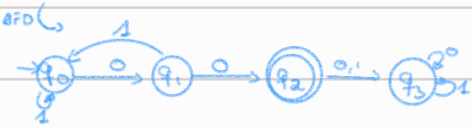
construir un autómata finito determinístico minimal que acepte el lenguaje $L_1 \setminus L_2$, a partir de autómatas que acepten L_1 y L_2.

3

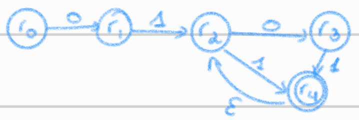
L_1:



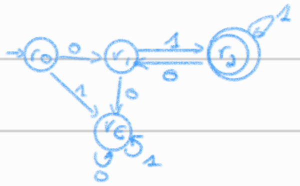
AFD L_1



L_2:

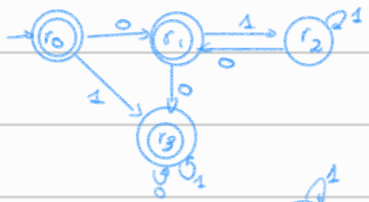


AFD L_2

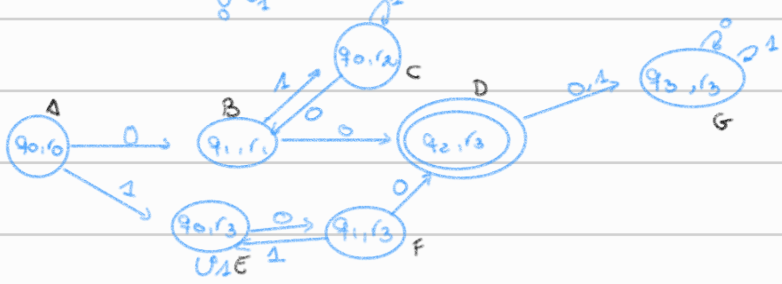


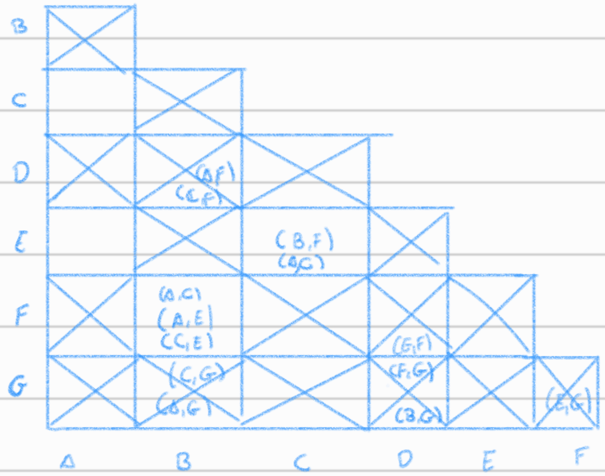
Sabemos por teoría de conjuntos que $L_1 \setminus L_2 = L_1 \cap \overline{L_2}$

$\overline{L_2}$:



$L_1 \cap \overline{L_2}$:





	0	1
B	C	D
A	B	E

Llegamos a que A, C, E son indistinguibles así como B y F también lo son

AFD minimal.

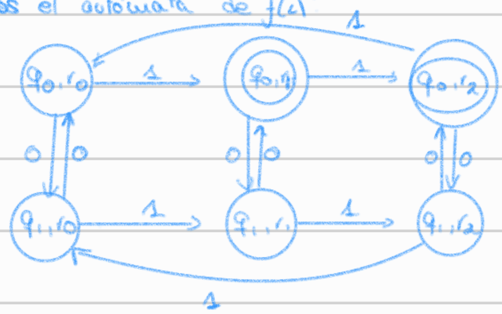
Solu

16. Dados los alfabetos $A = \{0, 1, 2, 3\}$ y $B = \{0, 1\}$ y el homomorfismo f de A^* en B^* dado por:

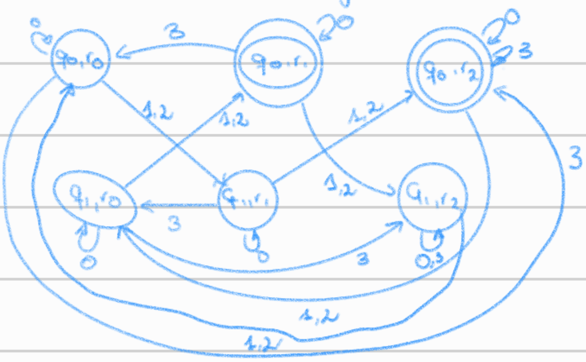
- $f(0) = 00, \quad f(1) = 01, \quad f(2) = 10, \quad f(3) = 11$

Sea L el conjunto de las palabras de B^* en las que el número de símbolos 0 es par y el de símbolos 1 no es múltiplo de 3. Construir un autómata finito determinista que acepte el lenguaje $f^{-1}(L)$.

Construimos el autómata de f(L):



Obtenemos el autómata de $f^{-1}(L)$.



18. Determinar si las expresiones regulares siguientes representan el mismo lenguaje:

a) $(b + (c + a)a^*(b + c))^*(c + a)a^*$

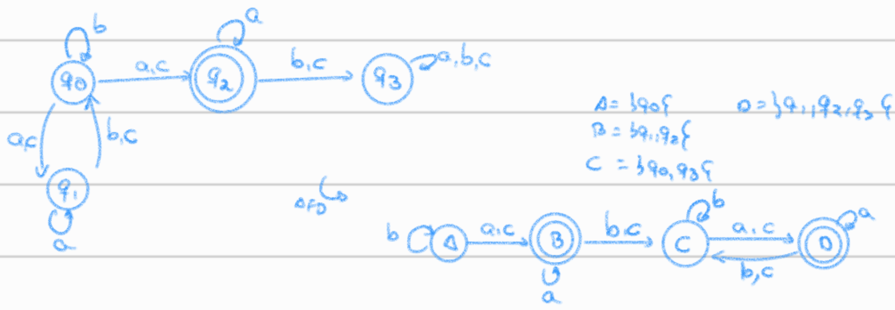
b) $b^*(c + a)((b + c)b^*(c + a))^*a^*$

c) $b^*(c + a)(a^*(b + c)b^*(c + a))^*a^*$

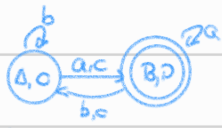
Justificar la respuesta.

Vamos a hallar los autómatas mínimos, he copiado los AFD para agilizar el proceso

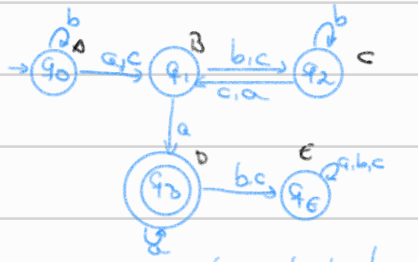
a)



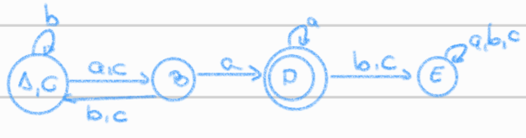
Aplicando el algoritmo conocido vemos que A y C son indistinguibles al igual que B, D



b)



De nuevo, aplicando el algoritmo llegamos a que A y C son indistinguibles



c)

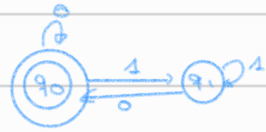
Donde vemos que A y C son indistinguibles

Como los AFD mínimos de a) y c) son el mismo entonces son equivalentes

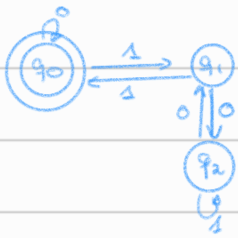
19. Construir un autómata finito determinista minimal que acepte el conjunto de palabras sobre el alfabeto $A = \{0, 1\}$ que representen números no divisibles por dos ni por tres.

Sea $L_1 = \{w \in \{0,1\}^* \mid w \neq 2k, k \in \mathbb{N}\}$ y $L_2 = \{w \in \{0,1\}^* \mid w \neq 3k, k \in \mathbb{N}\}$ entonces:

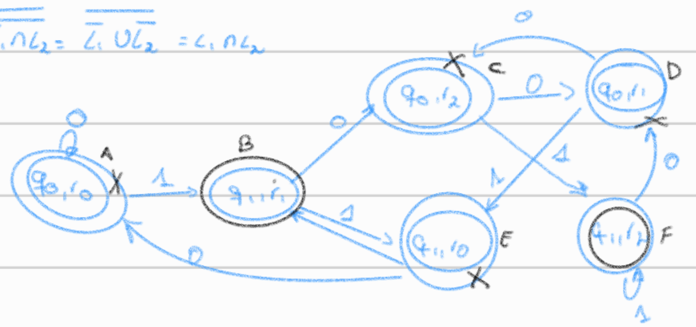
$\overline{L_1}$:



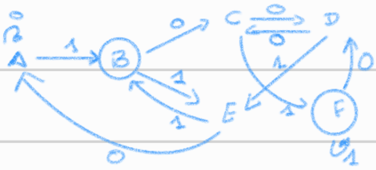
$\overline{L_2}$:

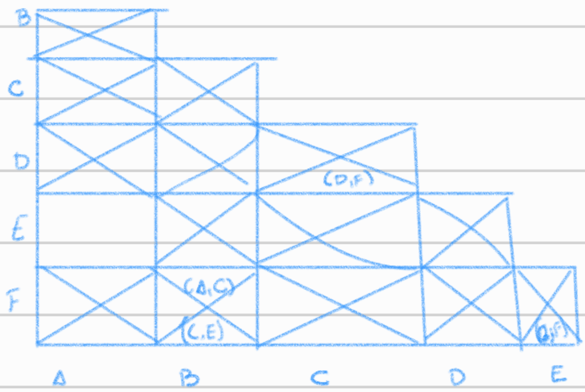


Usaremos que $\overline{\overline{L_1} \cap \overline{L_2}} = \overline{\overline{L_1} \cup \overline{L_2}} = L_1 \cap L_2$



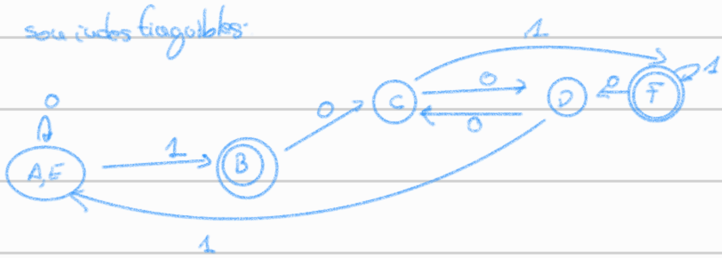
Minimizamos el autómata:





	0	1
A	A	B
C	D	F

Por tanto, A y E son indistinguibles:



22. ▪ Construye una gramática regular que genere el siguiente lenguaje:

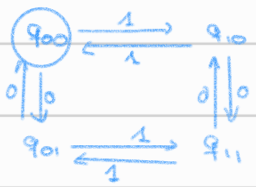
$$L_1 = \{u \in \{0,1\}^* \mid \text{ el número de 1's y el número de 0's en } u \text{ es par}\}$$

- Construye un autómata que reconozca el siguiente lenguaje:

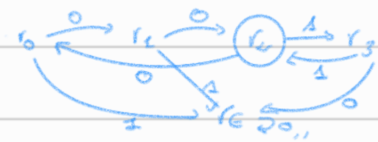
$$L_2 = \{0^n 1^m \mid n \geq 1, m \geq 0, n \text{ múltiplo de } 3, m \text{ par}\}$$

- Diseña el AFD mínimo que reconoce el lenguaje $(L_1 \cup L_2)$.

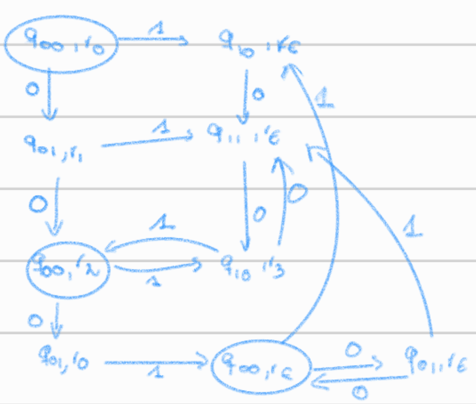
L_1:



L_2:



$L_1 \cup L_2$:



Habría que aplicar el algoritmo

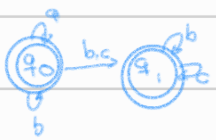
Si sale ⇒ comprobar antes

26. Construir autómatas finitos para los siguientes lenguajes sobre el alfabeto $\{a, b, c\}$:

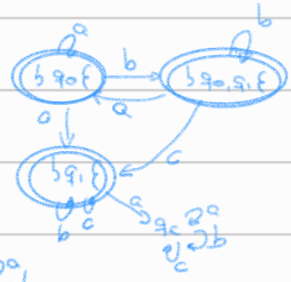
a) L_1: palabras del lenguaje $(\mathbf{a} + \mathbf{b})^*(\mathbf{b} + \mathbf{c})^*$.

b) L_2: palabras en las que nunca hay una $'a'$ posterior a una $'c'$.

a) AFND



AFD



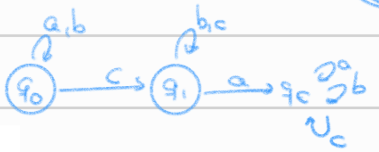
ca

$\{q_0\} = q_0$

$\{q_0, q_1\} = q_1$

$\{q_1\} = q_2$

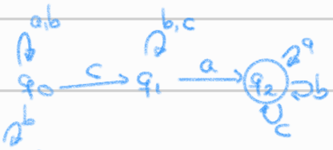
b)

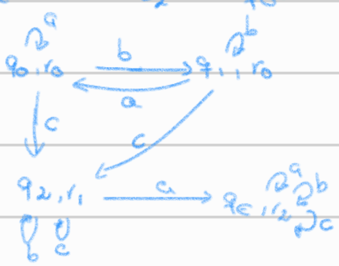


c) $(L_1 \setminus L_2) \cup (L_2 \setminus L_1)$

¿Qué podemos concluir sobre L_1 y L_2?

$L_1 \setminus L_2 = L_1 - L_2 = L_1 \cap \overline{L_2} \Rightarrow$ $\overline{L_2}$:



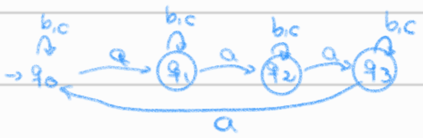


→ realmente no es así pero sirve igual

$L_2 \setminus L_1 = L_2 - L_1 = L_2 \cap \overline{L_1} \Rightarrow \overline{L_1} = \emptyset$ por tanto L_1 y L_2 son el mismo lenguaje

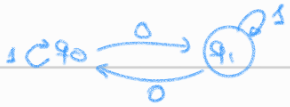
27. Si $f : \{0,1\}^* \to \{a,b,c\}^*$ es un homomorfismo dado por $f(0) = aab, f(1) = bbc$, dar autómatas finitos deterministas minimales para los lenguajes L y $f^{-1}(L)$ donde $L \subseteq \{a,b,c\}^*$ es el lenguaje en el que el número de símbolos a no es múltiplo de 4.

L:



que sabemos que es minimal

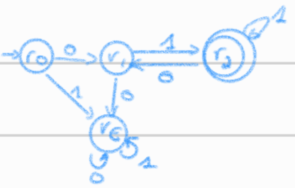
Como sabemos que $f(0) = aab$, $f(1) = bbc$ $\Rightarrow$ sabemos que no podemos tener un número par de ceros, por tanto:



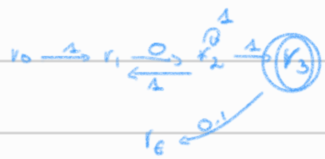
es el AFD minimal de $f^{-1}(L)$

28. Si L_1 es el lenguage asociado a la expresión regular $01(01+1)^*$ y L_2 el lenguaje asociado a la expresión $(1+10)^*01$, encontrar un autómata minimal que acepte el lenguaje $L_1 \setminus L_2$.

L_1:



L_2:



29. Sean los alfabetos $A_1 = \{a, b, c, d\}$ y $A_2 = \{0, 1\}$ y el lenguaje $L \subseteq A_2^*$ dado por la expresión regular $(\mathbf{0}+\mathbf{1})^*\mathbf{0}(\mathbf{0}+\mathbf{1})$, calcular una expresión regular para el lenguaje $f^{-1}(L)$ donde f es el homomorfismo entre A_1^* y A_2^* dado por

$$f(a) = 01, \qquad f(b) = 1, \qquad f(c) = 0, \qquad f(d) = 00$$

Salir

$$f^{-1}((0+1)^*0(0+1)) = \underbrace{(a+b+c+d)^*}_{0 \text{ y } 1 \text{ al gusto}} \underbrace{(cc+db+a+d)}_{00 \text{ ó } 01}$$

31. Dado el lenguaje L asociado a la expresión regular $(01 + 011)^*$ y el homomorfismo $f : \{0, 1\}^* \to \{0, 1\}^*$ dado por $f(0) = 01, f(1) = 1$, construir una expresión regular para el lenguaje $f^{-1}(L)$.

Salir

$$f^{-1}((01+011)^*) = (0+01)^*$$

Y es claramente correcto

32. Dar expresiones regulares para los siguientes lenguajes sobre el alfabeto $A_1 = \{0, 1, 2\}$:

a) L dado por el conjunto de palabras en las que cada 0 que no sea el último de la palabra va seguido por un 1 y cada 1 que no sea el último símbolo de la palabra va seguido por un 0.

b) Considera el homomorfimos de A_1 en $A_2 = \{0, 1\}$ dado por $f(0) = 001$, $f(1) = 100$, $f(2) = 0011$. Dar una expresión regular para $f(L)$.

c) Dar una expresión regular para LL^{-1}.

Ejercicio de otro tema, a) → construir autómata y resolver

b → aplicar f

c → obtener L^{-1} como autómata y resolver

33. Dados los lenguajes

Ejercicio de otro tema

$$L_1 = \{0^i 1^j \mid i \geq 1,\ j \text{ es par y } j \geq 2\}$$

y

$$L_2 = \{1^j 0^k \mid k \geq 1,\ j \text{ es impar y } j \geq 1\}$$

encuentra:

a) Una gramática regular que genere el lenguaje L_1.

b) Una expresión regular que represente al lenguaje L_2.

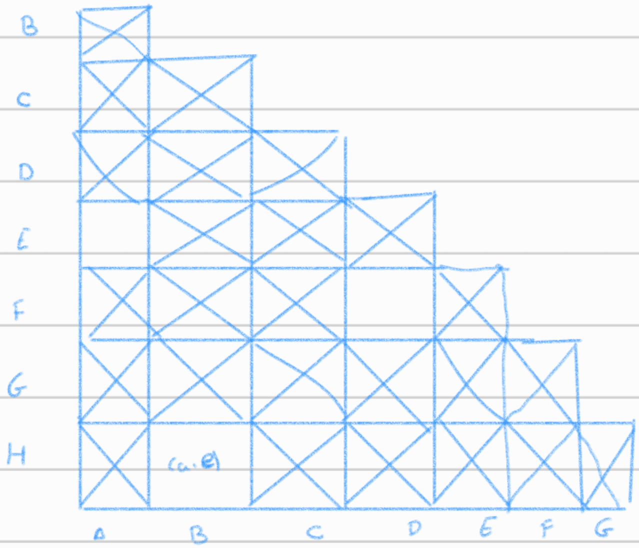
c) Un autómata finito determinista que acepte las cadenas de la concatenación de los lenguajes, L_1L_2. Aplica el algoritmo para minimizar este autómata.

34. Determinar si los siguientes autómatas finitos aceptan el mismo lenguaje justificando la respuesta (→ y ∗ indican el estado inicial y estado final respectivamente; los estados se indican con letras mayúsculas). Justificar la respuesta.

	0	1
→ A	B	F
B	G	C
*C	A	C
D	C	G
E	H	F
F	C	G
G	G	E
H	G	C

	0	1
→ A	G	C
B	B	A
C	D	B
*D	A	D
G	B	D

Para ello, vamos a aplicar el algoritmo de minimización en ambos



(A,C), (B,H) y (D,F) son indistinguibles

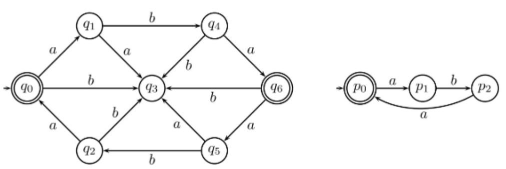
	0	1
→(A,C)	(B,H)	(D,F)
(B,H)	G	C
*C	(A,C)	C
(D,F)	C	G
G	G	(A,C)

Si nos fijamos bien, hemos obtenido el mismo autómata que el segundo; por tanto, es minimal. Además, tomando

$(A,C) \equiv A$, $(B,H) \equiv G$, $(D,F) \equiv C$, $C \equiv D$, $G \equiv B$

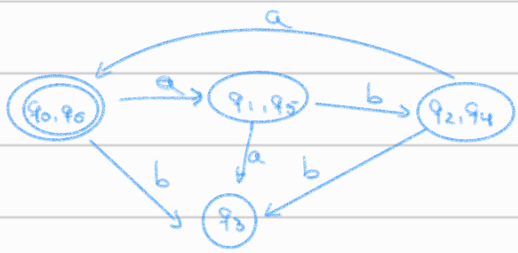
se ve todo más claro

35. Comprobar si los siguientes autómatas son equivalentes:



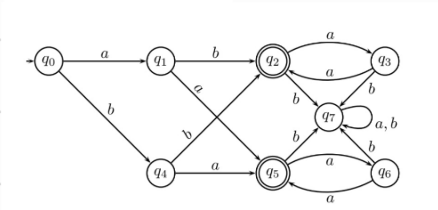
Para ello, aplicamos el algoritmo conocido sobre el primero de minimización de autómatas.

(q_1, q_5) son indistinguibles
(q_0, q_6) son indistinguibles
(q_2, q_4) son indistinguibles



Donde vemos que son equivalentes considerando a q_3 como estado de error no presente en el segundo autómata.

36. Minimizar el autómata:



Aplicamos el algoritmo de minimización.

1	X						
2	X	X					
3	X	X (0,2)	X				
4	X		X	X			
5	X	X (0,4)	(3,6)	X (1,2)	X (0,1)		
6	X	X	X	(2,5)	X	X	
7	X	X	(3,7) X (4,6)	X	X (9,3)	X (4,7) (6,7)	X
	0	1	2	3	4	5	6

(3,4) (4,1) (0,5) (1,6)

q_1, q_4 indistinguibles

q_3, q_6 indistinguibles

q_2, q_5 indistinguibles