Lema del bombeo

Sea L un conjunto regular, entonces existe $n \in \mathbb{N}$: $\forall z \in L$, si $|z| \geq n \Rightarrow z$ se puede expresar como $z = uvw$, donde:

- $|uv| \leq n$
- $|v| \geq 1$
- $(\forall i \geq 0)\ uv^i w \in L$

además, n puede ser el número de estados de cualquier autómata que acepte el lenguaje L.

NO suficiente

se trata de una $\boxed{\text{condición}}$ necesaria para los conjuntos regulares, luego, es útil para demostrar que un determinado lenguaje no es regular

Ver que un lenguaje no es regular

$\forall n \in \mathbb{N}$, existe una palabra $z \in L$, con $|z| \geq n$ tal que para toda descomposición $z = uvw$, si se verifica:

- $|uv| \leq n$
- $|v| \geq 1$

entonces,

$$\exists i \in \mathbb{N} : uv^i w \notin L$$

Ejemplo: Vamos a probar que $\{0^j 1^j : j \geq 0\}$ no es regular.

Veamos que, $\forall n \in \mathbb{N}$, existe una palabra $z \in L$, con $|z| \geq n$, $z = 0^n 1^n$, tal que, para toda descomposición $z = uvw$, si se verifica

- $|uv| \leq n$
- $|v| \geq 1$

entonces tenemos que $u = 0^k$, $v = 0^e$, $w = 0^{n-k-e} 1^n$, con $e \geq 1 \Rightarrow \exists i \in \mathbb{N} : uv^i w \notin L$. Basta con tomar $i = 2$ y vemos que $uv^2 w = 0^k 0^{2e} 0^{n-k-e} 1^n = 0^{n+e} 1^n \notin L \quad (e \geq 1)$

① Determinar si los siguientes lenguajes son regulares o libres de contexto. Justificar la respuesta:

a) $L = \{0^i b^j : i = 2j \text{ ó } 2i = j\}$

$\forall n \in \mathbb{N}$, existe una palabra $z \in L$, con $|z| \geq n$, $z = 0^n b^{2n}$ $(2n + n = 3n \geq n)$, tal que para toda descomposición $z = uvw$ se verifica:

- $|uv| \leq n$
- $|v| \geq 1$

Tenemos $\left.\begin{matrix} u = 0^m \\ v = 0^t \end{matrix}\right| \begin{matrix} (m+t \leq n) \\ (t \geq 1) \end{matrix}$ y $w = 0^{n-(m+t)} b^{2n} \Rightarrow \exists i \in \mathbb{N} : uv^i w \notin L$. Basta con tomar $i = 2$ y vemos que

$uv^2 w = 0^m 0^{2t} 0^{n-(m+t)} b^{2n} = 0^{n+t} b^{2n} \notin L \quad (t \geq 1)$, luego, el lenguaje no es regular.

d) $L = \{0^i 1^j 2^k : i = j \text{ ó } j = k\}$

$\forall n \in \mathbb{N}$, existe una palabra $z \in L$, con $|z| \geq n$, $z = 0^n 1^n 2^{3n}$ $(n + n + 3n = 5n \geq n)$ tal que para toda descomposición $z = uvw$, si se verifica:

- $|uv| \leq n$
- $|v| \geq 1$

entonces tenemos $\left.\begin{matrix} u = 0^m \\ v = 0^t \end{matrix}\right\} \begin{matrix} (m+t \leq n) \\ (t \geq 1) \end{matrix}$ y $w = 0^{n-(m+t)} 1^n 2^{3n} \Rightarrow \exists i \in \mathbb{N} : uv^i w \notin L$. Haciendo $i = 2$ tenemos que

$uv^2 w = 0^m 0^{2t} 0^{n-(m+t)} 1^n 2^{3n} = 0^{n+t} 1^n 2^{3n} \notin L \ (t \geq 1)$, luego, el lenguaje no es regular.

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

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

$\forall n \in \mathbb{N}$, existe una palabra $z \in L$, $|z| \geq n$, $z = 0^n 1^n 0 1^n 0^n \in L$ $(n + n + 1 + n + n = 4n + 1 \geq n)$ tal que para toda descomposición $z = uvw$ si se verifica:

- $|uv| \leq n$
- $|v| \geq 1$

entonces tenemos que $\begin{matrix} u = 0^k \\ v = 0^\ell \end{matrix} \Big| \begin{matrix} (k+\ell \le n) \\ (\ell \ge 1) \end{matrix}$ y $w = 0^{n-k-\ell} 1^n 0 1^n 0^n \Rightarrow \exists i \in \mathbb{N} : uv^i w \notin L$. Basta con tomar $i=2$ y vemos

que $uv^2w = 0^k 0^{2\ell} 0^{n-k-\ell} 1^n 0 1^n 0^n = 0^{n+\ell} 1^n 0 1^n 0^n \notin L \quad (\ell \ge 1)$. luego, el lenguaje no es regular.

b) Números en binario que sean múltiplos de 4.

Este lenguaje es regular y podemos construir un AFD asociado:

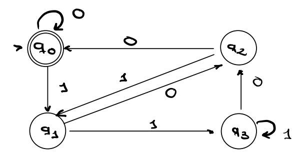
$q_0 \longrightarrow n \bmod 4 = 0$

$q_1 \longrightarrow n \bmod 4 = 1$

$q_2 \longrightarrow n \bmod 4 = 2$

$q_3 \longrightarrow n \bmod 4 = 3$

	0	1
q_0	q_0	q_1
q_1	q_2	q_3
q_2	q_0	q_1
q_3	q_2	q_3



La idea está en que meter un 0 en binario equivale a multiplicar por 2 $(101 = 1\cdot 2^2 + 0\cdot 2^1 + 1\cdot 2^0 \Rightarrow 1010 = 1\cdot 2^3 + 0\cdot 2^2 + 1\cdot 2^1 + 0\cdot 2^0 =$ Añade 0
$= (1\cdot 2^2 + 0\cdot 2^1 + 1\cdot 2^0) \times 2)$, mientras que meter un 1 equivale a meter un 0 (multiplicar por 2) y sumar 1. Luego, tomemos como ejemplo q_0. Si estamos en q_0 tenemos un múltiplo de 4, es decir, un número de la forma $4n$. Si añadimos un 0 (que es equivalente a multiplicar por 2 $\Rightarrow$ Tenemos $8n$ y como $8n \bmod 4 = 0 \Rightarrow$ seguimos en q_0. Si añadimos un uno es como multiplicar por 2 y sumar 1, luego tendremos $4n$ pasa a $8n+1$ y $8n+1 \bmod 4 = 1 \Rightarrow$ Pasamos a q_1. Ahora, si estamos en q_1 tenemos un número de la forma $4n+1$, si metemos 0 (es como multiplicar por 2) pasamos a $8n+2 \bmod 4 = 2 \Rightarrow q_2$ ... Así sucesivamente...

Operaciones sobre conjuntos regulares

- La unión de conjuntos regulares es regular: L_1, L_2 regulares $\Rightarrow L_1 \cup L_2$ regular.
- La concatenación de conjuntos regulares es regular: L_1, L_2 regulares $\Rightarrow L_1 L_2$ regular.
- La clausura de Kleene de un conjunto regular es regular: L regular $\Rightarrow L^*$ regular.
- El complementario de un conjunto regular es regular: Si $L \subseteq A^*$ regular $\Rightarrow \bar{L} = A^* \setminus L$ regular.
- La intersección de conjuntos regulares es regular: L_1, L_2 regulares $\Rightarrow L_1 \cap L_2$ regular.

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

Sea $L =$ " Palabras de $\{0,1\}^*$ que no contienen la subcadena 0110 " $\subseteq \{0,1\}^*$ y $L' =$ " Palabras de $\{0,1\}^*$ que contienen la subcadena 0110" $\subseteq \{0,1\}^*$.
L' es claramente regular y viene representado por la expresión $(0+1)^* 0110 (0+1)^*$ y como $L = \{0,1\}^* \setminus L' \Rightarrow L'$ también es regular.

3 Determinar qué lenguajes son regulares y qué lenguajes son libres de contexto de los siguientes:

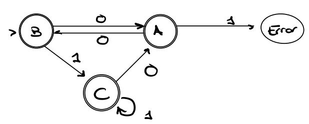
a) conjunto de palabras sobre el alfabeto $\{0,1\}$ en las que cada 1 va precedido por un número par de ceros

Es un lenguaje regular. Para demostrarlo podemos construir una gramática regular:

$S \rightarrow 0A \mid 1C \mid \varepsilon$

$A \rightarrow 0B \mid \varepsilon$

$C \rightarrow 1C \mid 0A \mid \varepsilon$



b) Conjunto $\{0^i 1^{2j} 0^{ij} : i,j \ge 0\}$

$\forall n \in \mathbb{N}$, existe una palabra $z \in L$, $|z| \ge n$, $z = 0^n 1^{2n} 0^{2n}$ $(2n+2n+n = 5n \ge n)$ tal que para toda descomposición $z = uv^iw$ si se verifica:

- $|uv| \le n$
- $|v| \ge 1$

entonces tenemos que $\begin{matrix} u = 0^m \\ v = 0^\ell \end{matrix} \Big| \begin{matrix} (m+\ell \le n) \\ (\ell \ge 1) \end{matrix}$ y $w = 0^{n-m-\ell} 1^{2n} 0^{2n} \Rightarrow \exists i \in \mathbb{N} : uv^i w \notin L$. Basta con tomar $i=2$

y vemos que $uv^2w = 0^m 0^{2\ell} 0^{n-m-\ell} 1^{2n} 0^{2n} = 0^{n+\ell} 1^{2n} 0^{2n} \notin L \quad (\ell \ge 1) \Rightarrow$ El lenguaje no es regular

c) conjunto $\{0^i 1^j 0^{ij} : i,j \ge 0\}$

$\forall n \in \mathbb{N}$, existe una palabra $z \in L$, $|z| \ge n$, $z = 0^n 1^n 0^{n^2}$ tal que para toda descomposición $z = uv^iw$ si se cumple:

- $|uv| \le n$
- $|v| \ge 1$

entonces tenemos que $\begin{matrix} u = 0^m \\ v = 0^\ell \end{matrix} \Big| \begin{matrix} (m+\ell \le n) \\ (\ell \ge 1) \end{matrix}$ y $w = 0^{n-m-\ell} 1^n 0^{n^2} \Rightarrow \exists i \in \mathbb{N} : uv^i w \notin L$. Basta con tomar $i=2$ y vemos que
$uv^2w = 0^m 0^{2\ell} 0^{n-m-\ell} 1^n 0^{n^2} = 0^{n+\ell} 1^n 0^{n^2} \notin L \, (\ell \ge 1) \Rightarrow$ El lenguaje no es regular.

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

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

$\forall n \in \mathbb{N}$, existe una palabra $z \in L$, $|z| \geq n$, $z = 0^{n+1} 1^n$, tal que para toda descomposición $z = uv^iw$ si se verifica que:

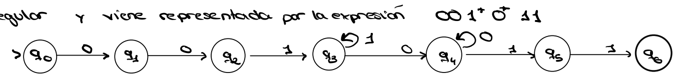
- $|uv| \leq n$
- $|v| \geq 1$

entonces tomando $\begin{matrix} u = 0^m \\ v = 0^\ell \end{matrix} \left| \begin{matrix} (m+\ell \leq n) \\ (\ell \geq 1) \end{matrix} \right.$ y $w = 0^{n-m-\ell} 0 1^n \Rightarrow \exists i \in \mathbb{N} : uv^iw \notin L$. Tomando $i = 0$ vemos que $uv^0w = 0^m 0^{n-m-\ell} 0 1^n =$

$= 0^{n-\ell+1} 1^n \notin L$ ($\ell \geq 1 \Rightarrow i = n-\ell+1 \leq n = j$) $\Rightarrow$ El lenguaje no es regular.

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

Este lenguaje es regular y viene representado por la expresión 001^+0^+11



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

Podemos expresar L_3 como concatenación de dos lenguajes $\Rightarrow L_3 = L_1 L_2$, donde $L_1 =$ " Lenguaje formado por la palabra 010 " y $L_2 =$ " Palabras de $\{0,1\}^*$ que no contienen la subcadena 010 ". L_1 es, evidentemente, regular. Por otro lado, podemos considerar el lenguaje $L_2' =$ " Palabras de $\{0,1\}^*$ que contienen la subcadena 010 ". L_2' es regular y viene representado por la expresión regular $(0+1)^* 010 (0+1)^*$ y, como $L_2 = \{0,1\}^* / L_2' \Rightarrow L_2$ regular y como la concatenación de conjuntos regulares es regular $\Rightarrow L_1L_2$ regular $\Rightarrow L_3$ regular.

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

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

no es regular.

$\forall n \in \mathbb{N}$, existe una palabra $z \in L$, $|z| \geq n$, $z = [1^n = 0^n + 1^n]$ $(n+n+n+2 = 3n+2 \geq n)$ tal que para toda descomposición $z = uv^iw$ si se verifica:

- $|uv| \leq n$
- $|v| \geq 1$

entonces tenemos que $\begin{matrix} u = 1^m \\ v = 1^\ell \end{matrix} \left| \begin{matrix} (m+\ell \leq n) \\ (\ell \geq 1) \end{matrix} \right.$ y $w = [1^{n-m-\ell} = 0^n + 1^n] \Rightarrow \exists i \in \mathbb{N} : uv^iw \notin L$. Basta con tomar $i = 2$ y vemos

que $uv^2w = [1^m 1^{2\ell} 1^{n-m-\ell} = 0^n + 1^n] = [\underbrace{1^{n+\ell}}_{x} = \underbrace{0^n}_{y} + \underbrace{1^n}_{z}] \notin L$ ($\ell \geq 1 \Rightarrow x \neq y + z \Rightarrow z \notin L$) $\Rightarrow$ El lenguaje no es regular.

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

a) $L = \{uvu^{-1} : u, v \in \{0,1\}^*\}$

$\forall n \in \mathbb{N}$, existe una palabra $z \in L$, $|z| \geq n$, $z = 0^n 1 0^n$, tal que para toda descomposición $z = uv^iw$ si se verifica

- $|uv| \leq n$
- $|v| \geq 1$

entonces, tomando $\begin{matrix} u = 0^m \\ v = 0^\ell \end{matrix} \left| \begin{matrix} (m+\ell \leq n) \\ (\ell \geq 1) \end{matrix} \right.$ y $w = 0^{n-m-\ell} 1 0^n \Rightarrow \exists i \in \mathbb{N} : uv^iw \notin L$. Basta con tomar $i = 2$ y vemos que

$uv^2w = 0^m 0^{2\ell} 0^{n-m-\ell} 1 0^n = \overbrace{0^{n+\ell}}^{u} 1 \overbrace{0^n}^{u^{-1}} \notin L$ ($\ell \geq 1 \Rightarrow n+\ell \neq n \Rightarrow |u| \neq |u^{-1}|$) $\Rightarrow$ El lenguaje no es regular.

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

$\forall n \in \mathbb{N}$, existe una palabra z, $|z| \geq n$, ; $z = \underbrace{0^n}_{u} 1 \underbrace{\underbrace{0^n}_{u^{-1}} 1^n}_{v}$ tal que para toda descomposición $z = uv^iw$, si se verifica:

- $|uv| \leq n$
- $|v| \geq 1$

entonces tenemos que, tomando $\begin{matrix} u = 0^m \\ v = 0^\ell \end{matrix} \left| \begin{matrix} (m+\ell \leq n) \\ (\ell \geq 1) \end{matrix} \right.$ y $w = 0^{n-m-\ell} 1 0^n 1^n \Rightarrow \exists i \in \mathbb{N} : uv^iw \notin L$. Basta con tomar $i = 2$ y vemos que $uv^2w = 0^m 0^{2\ell} 0^{n-m-\ell} 1 0^n 1^n = \underbrace{0^{n+\ell}}_{u} 1 \underbrace{0^n}_{u^{-1}} 1^n \notin L$ ($\ell \geq 1 \Rightarrow n+\ell \neq n \Rightarrow |u| = |u^{-1}|$) $\Rightarrow$ No es regular.

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

Este lenguaje es regular, y viene representado por la expresión $(0+1)^* 1 (0+1)(0+1)$

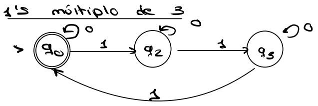
Autómata producto para la unión e intersección

Para construir un autómata que acepte la unión de los lenguajes aceptados por dos autómatas con el autómata producto, basta con hacer finales las parejas de estados en las que, al menos uno de los dos, es final. El de la intersección es igual, pero para marcar una pareja como final, ambos estados deben ser finales.

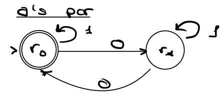
⑦ Obtener autómatas finitos deterministicos para los siguientes lenguajes sobre el alfabeto $\{0,1\}$.

a) Palabras en las que el número de 1 es múltiplo de 3 y el número de 0 es par.

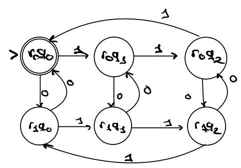
1's múltiplo de 3



0's par

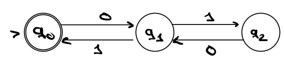


Autómata producto (intersección):



b) $\{(01)^{2i} : i \geq 0\}$

Es regular y viene dado por el autómata:



c) $\{(0^{2i}1^{2i}) : i \geq 0\}$

$\forall n \in \mathbb{N}$, existe una palabra $z \in L$, $|z| \geq n$, $z = 0^{2n}1^{2n}$, tal que para toda descomposición $z = uv^iw$ si verifica:

- $|uv| \leq n$
- $|v| \geq 1$

entonces, tomando $\left.\begin{array}{l} u = 0^m \\ v = 0^{\ell} \end{array}\right\} \begin{array}{l} (m+\ell \leq n) \\ (\ell \geq 1) \end{array}$ y $w = 0^{2n-m-\ell}1^{2n} \Rightarrow \nexists i \in \mathbb{N} : uv^iw \notin L$. Basta con tomar $i=2$ y vemos que

$uv^2w = 0^m0^{2\ell}0^{2n-m-\ell}1^{2n} = 0^{2n+\ell}1^{2n} \notin L$ $(\ell \geq 1 \Rightarrow 2n+\ell \neq 2n)$ $\Rightarrow$ El lenguaje no es regular.

Autómata Minimal

Un autómata finito determinista M se dice minimal si no existe otro autómata con menos estados que él y que acepte el mismo lenguaje.

- Si $M = (Q, A, \delta, q_0, F)$ es un autómata finito determinista y p, q son dos estados de Q, decimos que p y q son indistinguibles si y solo si se cumple que $\forall u \in A^*$, $(\delta^*(p,u) \in F \Leftrightarrow \delta^*(q,u) \in F)$. En caso contrario, diremos que son distinguibles. Para que p, q sean distinguibles debe existir $u \in A^*$ tal que el conjunto $\{\delta^*(p,u), \delta^*(q,u)\}$ haya un estado final y otro no final.
- La relación de indistinguibilidad es una relación de equivalencia en el conjunto Q de estados (reflexiva, simétrica y transitiva).
- Un estado final y uno no final son siempre distinguibles: al leer la palabra vacía en un caso se llega a un estado final y en otro a un estado no final.
  - Si $\underset{(u \in A^*)}{a \in A}$, y los estados $\underset{(\delta^*(p,u))}{\delta(p,a)}$ y $\underset{(\delta^*(p,u))}{\delta(q,a)}$ son distinguibles $\Rightarrow$ p y q distinguibles.
  - Si p y q son dos estados distinguibles $\Rightarrow \underset{(\forall u \in A^*)}{\forall a \in A}$, los estados $\underset{(\delta^*(p,u))}{\delta(p,a)}$ y $\underset{(\delta^*(q,u))}{\delta(q,a)}$ son indistinguibles

Si partimos de un autómata $M = (Q, A, \delta, q_0, F)$, sea R la relación de equivalencia de indistinguibilidad entre estados, $[q]$ la clase de equivalencia asociada al estado q. El nuevo autómata que agrupa los estados indistinguibles es $M_m = (Q_m, A, \delta_m, q_0^m, F_m)$

- $Q_m = \{[q] : q \text{ accesible desde } q_0\}$
- $\delta_m([q], a) = [\delta(q,a)]$
- $F_m = \{[q] : q \in F\}$
- $q_0^m = [q_0]$

Se verifica que ambos autómatas aceptan el mismo lenguaje, $L(M) = L(M_m)$

Teorema: Si $M = (Q, A, \delta, q_0, F)$ y $M' = (Q', A, \delta', q_0', F')$ aceptan el mismo lenguaje y $u, v \in A^*$ son tales que $\delta^*(q_0, u) = \delta^*(q_0, v)$, entonces $q_i' = \delta'^*(q_0', u)$ y $q_j' = \delta'^*(q_0', v)$ son indistinguibles en M'.

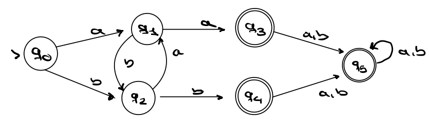
- Un autómata sin estados inaccesibles es minimal $\Leftrightarrow$ No tiene una pareja de estados indistinguibles.

Teorema de unicidad: Si $M = (Q, A, \delta, q_0, F)$ y $M' = (Q', A, \delta', q_0', F')$ son dos autómatas minimales que aceptan el mismo lenguaje, entonces son isomorfos, es decir, existe una aplicación biyectiva $f : Q \to Q'$ tal que $f(q_0) = q_0'$, $f(F) = F'$ y si $\delta(q,a) = p \Rightarrow \delta'(f(a), a) = f(p)$

Cálculo

1) Eliminar estados inaccesibles
2) Para cada pareja de estados $\{q_i, q_j\}$, si uno de ellos es final y otro no, hacer la variable booleana (indestinguible) igual a true.
3) Para cada pareja de estados $\{q_i, q_j\}$ y para cada símbolo, a, del alfabeto, calcular los estados q_k y q_ℓ a los que evolucionan q_i y q_j leyendo a. Si $q_k \neq q_\ell$, entonces miramos si $\{q_k, q_\ell\}$ está marcado. En ese caso marcamos $\{q_i, q_j\}$ y, recursivamente todas las parejas de la lista asociada. Si $\{q_k, q_\ell\}$ no está marcada, se añade $\{q_i, q_j\}$ a la lista asociada a la pareja $\{q_k, q_\ell\}$.

10 Encontrar un AFD minimal para el lenguaje $(a+b)^*(aa+bb)(a+b)^*$



·) Todos los estados son accesibles

q_1	X				
q_2	X	X			
q_3	X	X	X		
q_4	X	X	X		
q_5	X	X	X		
	q_0	q_1	q_2	q_3	q_4

2 Si $\{q_k, q_e\}$ está marcada, entonces se marca $\{q_i, q_j\}$ y recursivamente todas las parejas de la lista asociada

1 Marcamos las parejas de estados finales y no finales

3 Están en la misma clase

	a	b
q_0	q_1	q_2
q_1	q_3	q_2

Está marcada por 1

	a	b
q_0	q_1	q_2
q_2	q_1	q_4

	a	b
q_1	q_3	q_2
q_2	q_1	q_4

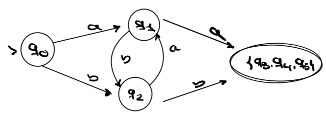
	a	b
q_3	q_5	q_5
q_4	q_5	q_5

	a	b
q_3	q_5	q_5
q_5	q_5	q_5

	a	b
q_4	q_5	q_5
q_5	q_5	q_5

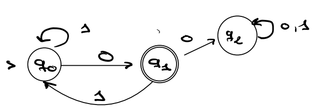
Luego, tenemos $[q_3]=[q_4]$ y que $[q_4]=[q_5] \Rightarrow [q_3]=[q_4]=[q_5]$

Por lo que, el autómata minimal sería:

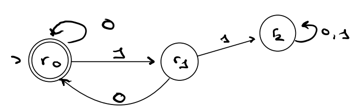


8 Dar una expresión regular para la intersección de los lenguajes asociados a las expresiones regulares $(01+1)^*0$ y $(10+0)^*$. Se valorará que se construya el autómata que acepta la intersección de estos lenguajes, se minimice y, a partir del resultado, se construya la expresión regular.

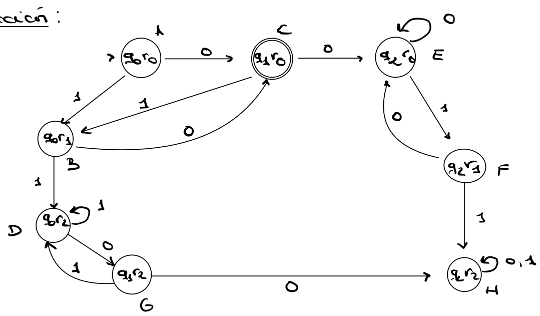
·) $(01+1)^*0$ ⟶ No acepta dos 0 consecutivos



·) $(10+0)^*$ ⟶ No acepta dos unos consecutivos



Intersección:



1) Quitamos estados no accesibles ⟶ No hay

2) Marcamos parejas de estados finales y no finales

B	X						
C	X	X					
D	X	X	X				
E	X	X	X				
F	X	X	X				
G	X	X	X				
H	X	X	X				
	A	B	C	D	E	F	G

	0	1
A	C	B
B	C	D

Por recursividad

	0	1
D	G	D
A	C	B

	0	1
D	G	D
B	C	D

	0	1
E	E	F
A	C	B

	0	1
E	E	F
B	C	D

	0	1
E	E	F
D	G	D

	0	1
F	E	H
A	C	B

	0	1
F	E	H
B	C	D

	0	1
F	E	H
D	G	D

	0	1
F	E	H
E	E	F

	0	1
G	H	D
A	C	B

	0	1
G	H	D
B	C	D

	0	1
G	H	D
D	G	D

	0	1
G	H	D
E	E	F

	0	1
G	H	D
F	E	H

	0	1
H	H	H
A	C	B

	0	1
H	H	H
B	C	D

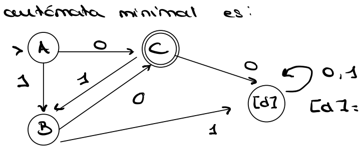
	0	1
H	H	H
D	G	D

	0	1
H	H	H
E	E	F

	0	1
H	H	H
F	E	H

	0	1
H	H	H
G	H	D

Luego, el autómata minimal es:



$[d]: [D]=[E]=[F]=[G]=[H]$

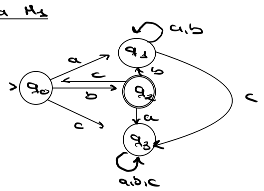
13 determinar autómatas minimales para los lenguajes $L(M_1) \cup L(M_2)$ y $L(M_1) \cap L(\overline{M_2})$, donde:

- $M_1 = (\{q_0, q_1, q_2, q_3\}, \{a,b,c\}, \delta_1, q_0, \{q_2\})$
- $M_2 = (\{q_0, q_1, q_2, q_3\}, \{a,b,c\}, \delta_2, q_0, \{q_2\})$

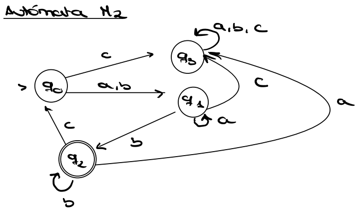
δ_1	q_0	q_1	q_2	q_3
a	q_1	q_1	q_3	q_3
b	q_2	q_1	q_1	q_3
c	q_3	q_3	q_0	q_3

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

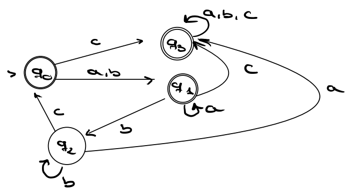
Autómata M_1



Autómata M_2



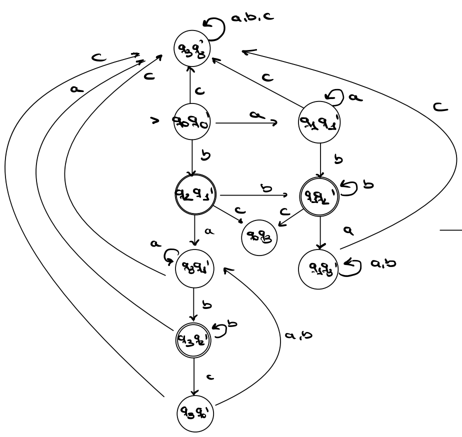
Autómata $\overline{M_2}$



Si $M = (Q, A, \delta, q_0, F)$ es un autómata finito determinista que acepta el lenguaje L, entonces $M' = (Q, A, \delta, q_0, Q \setminus F)$ acepta $A^* \setminus L$.

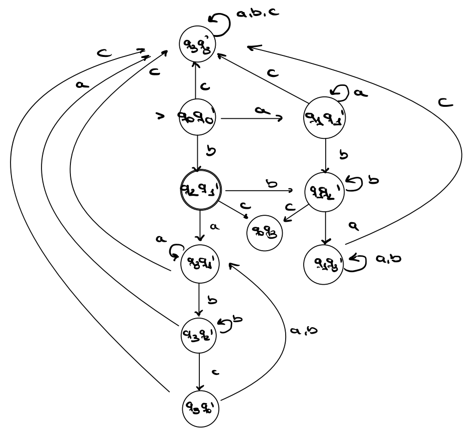
Autómata $L(M_1) \cup L(M_2)$ (Autómata producto para la unión):

Renombramos en M_2 $\{q_0, q_1, q_2, q_3\} \longrightarrow \{q_0', q_1', q_2', q_3'\}$



Para $L(M_1) \cap L(\overline{M_2})$ basta con cambiar los estados finales

Revisar

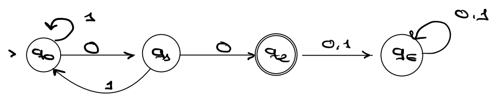


15 Sean los lenguajes:

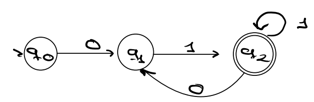
- $L_1 = (01+1)^* 00$
- $L_2 = 01(01+1)^*$

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

Autómata L_1



Autómata L_2:



Notemos que las palabras de L_1 acaban en 00, mientras que las de L_2 acaban en 1 $\Rightarrow L_1 \cap L_2 = \emptyset \Rightarrow L_1 \setminus L_2 = L_1$. Por lo que tomamos el autómata de L_1 y lo transformamos a minimal.

1) Eliminamos estados inaccesibles

En este caso podemos eliminar q_E.

2) Marcamos parejas de estados finales y no finales.

q_1	X	
q_2	X	X
	q_0	q_1

	0	1
q_1	q_2	q_0
q_0	q_1	q_0

$\Rightarrow$ Ya tenemos el autómata minimal

Homomorfismo

Si A y B son alfabetos y $f: A^* \longrightarrow B$ un homomorfismo entre ellos, entonces si $L \subseteq A^*$ es un lenguaje regular, $f(L) = \{ f(u) \in B^* : u \in L \}$ es también un lenguaje regular

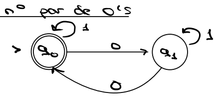
16 Dados los alfabetos $A = \{0,1,2,3\}$, $B = \{0,1\}$ y el homomorfismo f de A^* en B^* dado por:

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

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

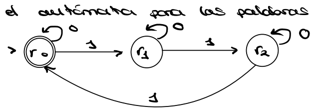
$L = \{ u \in B^* : n^o$ 0's es par y n^o 1's no es múltiplo de 3 $\}$

Autómata n^o par de 0's

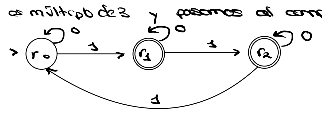


Autómata n^o de 1's no es múltiplo de 3

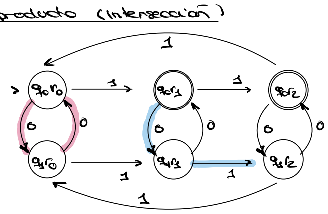
Hallamos el autómata para las palabras de B^* donde el n^o de 1's es múltiplo de 3 y pasamos al complementario:



$\Longrightarrow$ complementario



Autómata producto (intersección)



Autómata $f^{-1}(L)$

$f^{-1}: B^* \longrightarrow A^*$

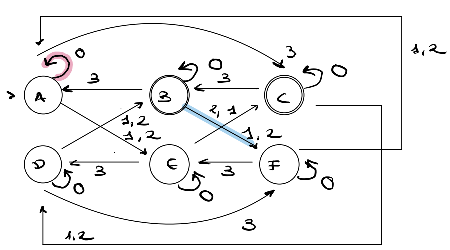
$f^{-1}(00) = 0$

$f^{-1}(01) = 1$

$f^{-1}(10) = 2$

$f^{-1}(11) = 3$

$\Rightarrow$

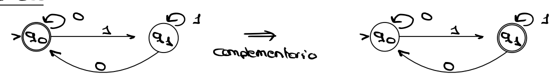


(19) Construir un autómata finito determinista minimal que acepte el conjunto de palabras sobre el alfabeto $A = \{0,1\}$ que representa números no divisibles ni por dos ni por tres.

Autómata que no acepta números divisibles por dos

$q_0 \longrightarrow n \bmod 2 = 0$

$q_1 \longrightarrow n \bmod 2 = 1$

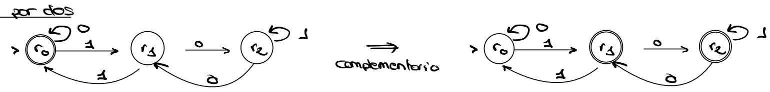


Autómata que no acepta números divisibles por dos

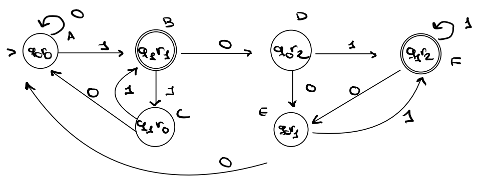
$r_0 \longrightarrow n \bmod 3 = 0$

$r_1 \longrightarrow n \bmod 3 = 1$

$r_2 \longrightarrow n \bmod 3 = 2$



Autómata producto de la intersección



1) Eliminamos estados inaccesibles ⤳ No hay

2) Marcamos las parejas de estados finales y no finales

B	x				
C		x			
D	x	x	x		
E	x	x	x	x	
F	x	x	x	x	x
	A	B	C	D	E

	0	1
C	A	B
A	A	B

	0	1
D	E	F
A	A	B

	0	1
D	E	F
C	A	B

→ Recursivamente

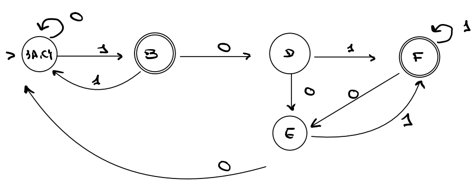
	0	1
E	A	F
A	A	B

	0	1
E	A	F
C	A	B

	0	1
E	A	F
D	E	F

	0	1
F	E	F
B	D	C

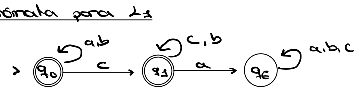
Luego, tenemos que $C \equiv A$ ($[C] = [A]$) :



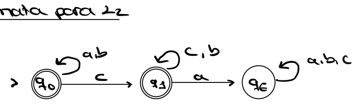
(26) Construir autómatas finitos para los siguientes lenguajes sobre el alfabeto $\{a,b,c\}$:

- L_1 = " Palabras del lenguaje $(a+b)^*(b+c)^*$ "
- L_2 = " Palabras en las que hay una 'a' posterior a una 'c' "
- $(L_1 \setminus L_2) \cup (L_2 \setminus L_1)$

Autómata para L_1



Autómata para L_2



$L_1 = L_2 \Rightarrow L_1 \setminus L_2 = L_2 \setminus L_1 = \emptyset$

33 Dados los lenguajes

- $L_1 = \{0^i 1^j : i \geq 1,\ j \text{ es par},\ j \geq 2\}$
- $L_2 = \{1^j 0^k : k \geq 1,\ j \text{ es impar},\ j \geq 1\}$

encuentra:

a) una gramática regular que genere L_1 y una expresión regular que represente L_2.

Gramática Regular L_1

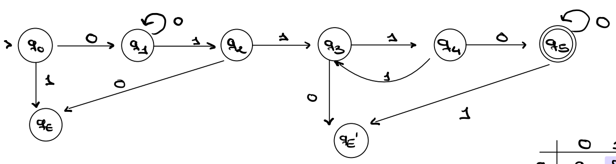
$S \rightarrow 0S \mid 0A$

$A \rightarrow 11A \mid 11$

Expresión Regular L_2

$1(11)^* 0^+$

b) un autómata finito determinista que acepte la concatenación de los lenguajes $L_1 L_2$ y minimízalo.



1) Eliminamos estados inaccesibles

2) Marcamos las parejas de estados finales y no finales

q_1	X				
q_2	X	X			
q_3	X	X	X		
q_4	X	X	X	X	
q_5	X	X	X	X	X
	q_0	q_1	q_2	q_3	q_4

Ya es minimal

	0	1
q_0	q_1	q_E
q_1	q_1	q_2

	0	1
q_2	q_E	q_3
q_0	q_1	q_E

	0	1
q_2	q_E	q_3
q_1	q_1	q_2

Los estados de error siempre son distinguibles ya que no pueden generar palabras porque no llegan a un estado final

	0	1
q_3	q_E'	q_4
q_0	q_1	q_E

	0	1
q_3	q_E'	q_4
q_1	q_1	q_2

	0	1
q_3	q_E'	q_4
q_2	q_E	q_3

	0	1
q_4	q_5	q_3
q_0	q_1	q_E

	0	1
q_4	q_5	q_3
q_1	q_1	q_2

	0	1
q_4	q_5	q_3
q_2	q_E	q_3

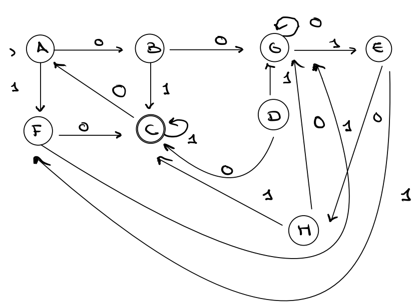
	0	1
q_4	q_5	q_3
q_3	q_E'	q_4

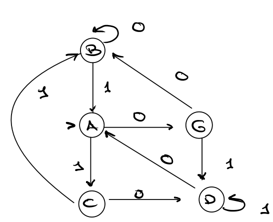
34 Justificar si los siguientes autómatas finitos aceptan el mismo lenguaje justificando la respuesta ($\rightarrow$ y $*$ indican el estado inicial y estado final respectivamente, los dos estados se indican con letras mayúsculas). Justificar la respuesta:

	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

Comenzamos construyendo los autómatas:





Vamos a comenzar minimizando el primer autómata:

1) Eliminamos estados inaccesibles ⤳ No hay

2) Marcamos las parejas de estados finales y no finales

B	x						
C	x	x					
D	x	x	x				
E		x	x	x			
F	x	x	x		x		
G	x	x	x	x	x	x	
H	x		x	x	x	x	x
	A	B	C	D	E	F	G

	0	1
B	G	C
A	B	F

	0	1
D	C	G
A	B	F

	0	1
D	C	G
B	G	C

	0	1
E	H	F
A	B	F

	0	1
E	H	F
B	G	C

	0	1
E	H	F
D	C	G

	0	1
F	C	G
A	B	F

	0	1
F	C	G
B	G	C

	0	1
F	C	G
D	C	G

	0	1
F	C	G
E	H	F

	0	1
G	G	E
A	B	F

	0	1
G	G	E
B	G	C

	0	1
G	G	E
D	C	G

	0	1
G	G	E
E	H	F

	0	1
G	G	E
F	C	G

	0	1
H	G	C
A	B	F

	0	1
H	G	C
B	G	C

	0	1
H	G	C
D	C	G

	0	1
H	G	C
E	H	F

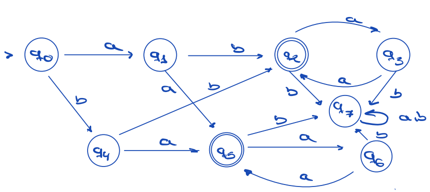
	0	1
H	G	C
F	C	G

	0	1
H	G	C
G	G	E

Luego $E \equiv A$, $B \equiv H$, $D \equiv F$

Si simplificamos el autómata, vemos que son los mismos

(5) Minimizar el autómata



1) No hay estados inaccesibles:

2) Hacemos las parejas de estados finales y no finales

q_1	x						
q_2	x	x					
q_3	x	x	x				
q_4	x		x	x			
q_5	x	x		x	x		
q_6	x	x	x		x	x	
q_7	x	x	x	x	x	x	x
	q_0	q_1	q_2	q_3	q_4	q_5	q_6

	a	b
q_1	q_5	q_2
q_0	q_1	q_4

	a	b
q_3	q_2	q_7
q_0	q_1	q_4

	a	b
q_3	q_2	q_7
q_1	q_5	q_2

	a	b
q_4	q_5	q_2
q_0	q_1	q_4

	a	b
q_4	q_5	q_2
q_1	q_5	q_2

	a	b
q_4	q_5	q_2
q_3	q_2	q_7

	a	b
q_5	q_6	q_7
q_2	q_3	q_7

	a	b
q_6	q_5	q_7
q_0	q_1	q_4

	a	b
q_6	q_5	q_7
q_1	q_5	q_2

	a	b
q_6	q_5	q_7
q_3	q_2	q_7

	a	b
q_6	q_5	q_7
q_4	q_0	q_2

	a	b
q_7	q_7	q_7
q_0	q_1	q_4

	a	b
q_7	q_7	q_7
q_1	q_5	q_2

	a	b
q_7	q_7	q_7
q_3	q_2	q_7

	a	b
q_7	q_7	q_7
q_4	q_5	q_2

	a	b
q_7	q_7	q_7
q_6	q_5	q_7

Luego, tenemos que $q_6 \equiv q_3$, $q_5 \equiv q_2$, $q_4 \equiv q_1$, por lo que el autómata mínimo será: