

Modelos de Computación:

Relación de problemas 3

David Cabezas Berrido

Ejercicio 27. Expresiones regulares para los lenguajes sobre $\{0, 1\}$:

a) Palabras con número de símbolos 0 múltiplo de 3.

$$(1^*01^*01^*01^*)^*$$

b) Palabras que contienen como subcadena a 1100 ó a 00110.

$$(0 + 1)^*(1100 + 00110)(0 + 1)^*$$

c) Palabras en las que cada cero forma parte de una subcadena de 2 ceros y cada 1 forma parte de una subcadena de 3 unos.

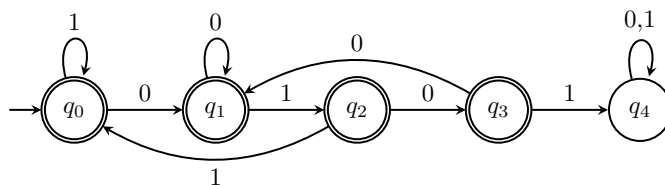
$$(00 + 111)^*$$

d) Palabras en las que el número de ocurrencias de la subcadena 011 es menor o igual que el de ocurrencias de la subcadena 110.

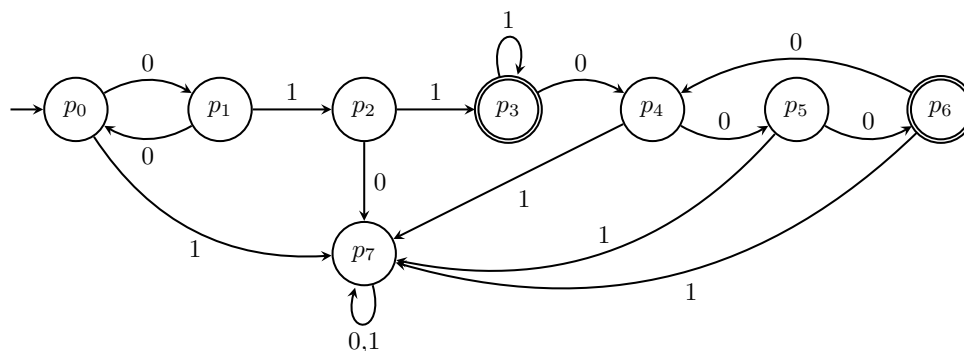
Este lenguaje no es regular, luego no existe una expresión regular para el mismo.

Ejercicio 29. Encontrar gramática tipo 3 o autómata finito que reconozca el lenguaje.

- $L_1 = \{u \in \{0, 1\}^* : u \text{ no contiene la subcadena '0101'}\}$

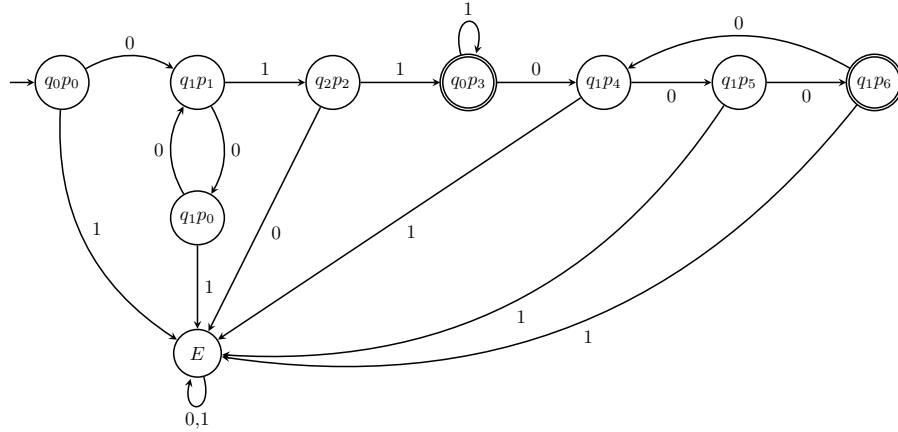


- $L_2 = \{0^i1^j0^k : i \geq 1, k \geq 0, i \text{ impar}, k \text{ múltiplo de 3 y } j \geq 2\}$

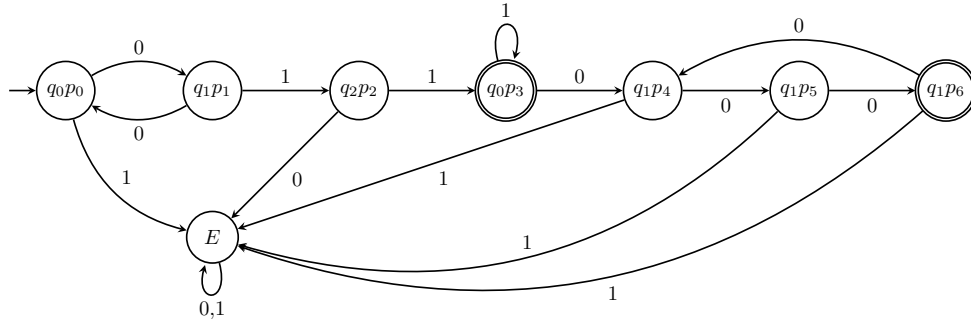


Diseñar el AFD minimal que reconoce $(L_2 \cap L_1)$.

El autómata sin minimizar (sólo agrupando los estados de error) es el siguiente.



Tiene sólo dos estados indistinguibles: q_0p_0 y q_1p_0 , luego el autómata minimal es el que resulta de agrupar ambos estados.



Podemos apreciar que este autómata es isomorfo al que hemos diseñado para L_2 , lo cual ocurre porque $L_2 \subset L_1$ (es claro que las palabras de L_2 no tienen a '0101' como subcadena).

Ejercicio 45. $A = \{0, 1, +, =\}$, probar que el lenguaje

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

no es regular.

Probaré que ADD no satisface el lema de bombeo. Sea $n \in \mathbb{N}$, la palabra $1^n = 0 + 1^n$ está en el lenguaje y tiene longitud $2n + 3 \geq n$. Tomamos una descomposición cualquiera de la forma uvw con $|uv| = l \leq n$ y $|v| \geq 1$, por lo tanto $v = 1^k$ para algún $k \geq 1$.

La palabra uv^2w será $1^{l-k}1^{2k}1^{n-l} = 0 + 1^n$, que simplificando queda $1^{n+k} = 0 + 1^n$. Como $k \geq 1$, el valor numérico de 1^{n+k} será estrictamente mayor que el de 1^n , por lo tanto la palabra uv^2w no está en ADD (la suma no es correcta).

Resumiendo, dado un valor de n cualquiera, podemos encontrar en el lenguaje una palabra z de longitud mayor que n tal que para toda descomposición de z de la forma uvw , la palabra uv^2w no está en el lenguaje. Luego ADD no es regular.

Ejercicio 46. La mezcla perfecta de dos lenguajes L_1 y L_2 sobre un alfabeto A se define como

$$\{w \mid w = a_1b_1a_2b_2 \dots a_kb_k \text{ donde } a_1 \dots a_k \in L_1, b_1 \dots b_k \in L_2, a_i, b_i \in A\}$$

Demostrar que si L_1 y L_2 son regulares, su mezcla perfecta lo es.

Consideremos autómatas finitos deterministas que reconocen a L_1 y a L_2

$$M_1 = (Q, A, q_0, \delta_1, F)$$

$$M_2 = (P, A, p_0, \delta_2, G)$$

Definimos el autómata

$$M = (Q \times P \times \{0, 1\}, A, (q_0, p_0, 0), \delta, F \times G \times \{0\})$$

donde

$$\delta((q, p, 0), a) = (\delta_1(q, a), p, 1) \quad \forall (q, p) \in Q \times P, a \in A$$

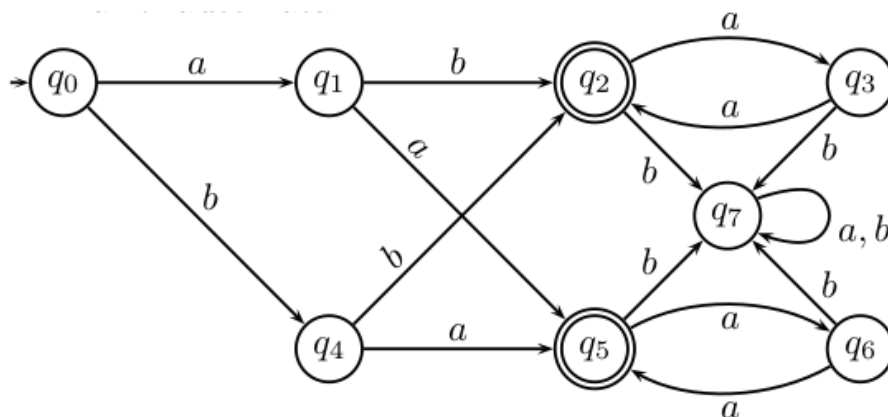
$$\delta((q, p, 1), a) = (q, \delta_2(p, a), 0) \quad \forall (q, p) \in Q \times P, a \in A$$

Una palabra $w = a_1a_2 \dots a_{2k} \in A^*$ lleva al estado $(\delta_1(\varepsilon, u), \delta_2(\varepsilon, v), 0)$ donde $u = a_1a_3 \dots a_{2k-1}$ y $v = a_2a_4 \dots a_{2k}$, por lo que $w \in L(M) \Leftrightarrow (\delta_1(\varepsilon, u), \delta_2(\varepsilon, v), 0) \in F \times G \times \{0\} \Leftrightarrow u \in L_1$ y $v \in L_2$. Luego M acepta una palabra de longitud par si y sólo si pertenece a la mezcla perfecta de L_1 y L_2 .

Una palabra de longitud impar llevará a un estado de $Q \times P \times \{1\}$, luego M no aceptará palabras de longitud impar, puesto que estas nunca estarán en la mezcla perfecta de L_1 y L_2 .

Resumiendo, hemos hallado un AFD que reconoce la mezcla perfecta de L_1 y L_2 , luego es regular.

Ejercicio 47. Minimizar el autómata.



No hay estados inaccesibles.

Usando el algoritmo visto en clase, he obtenido que las parejas de estados indistinguibles son (q_1, q_4) , (q_2, q_5) y (q_3, q_6) .

El autómata minimal será el resultante de agrupar cada una de las parejas.

