# Teoría Matemática de la Computación
# Segundo Problemario

Prof. Miguel A. Pizaña

19 de Agosto de 2008

1. Un lenguaje $L$ es *cofinito* si su complemento $\bar{L} = \Sigma^* - L$ es finito. Demuestre que todo lenguaje cofinito es regular.

2. Demuestre, para cualquier autómata $M$, con función de transición $\delta$ se tiene que $\delta(q, uv) = \delta(\delta(q, u), v)$.

3. Demuestre el lema siguiente. Muestre que el lema también vale si el conjunto cadenas $X := \{x_1, x_2, \ldots\}$ es infinito.

   **Lema** Si $L$ es un lenguaje regular y suponga que existen cadenas $x_1, x_2, \ldots, x_n, z_{12}, z_{13}, z_{23}, \ldots, z_{ij}, \ldots z_{(n-1)n} \in \Sigma^*$ que satisfacen: ($x_i z_{ij} \in L$ y $x_j z_{ij} \notin L$) ó ($x_i z_{ij} \notin L$ y $x_j z_{ij} \in L$). Entonces, cualquier AFD que reconozca a $L$ debe tener almenos $n$ estados.

4. Muestre que cualquier AFD que reconozca a $L = (0+1)^*0(0+1)^n$ necesita al menos $2^{n+1}$ estados.

5. Enuncie y demuestre el Lema alfa.

6. Determinar si los siguientes lenguajes son regulares o no (Encuentre un AFD que lo reconozca o use el lema alfa).

   (a) $\{\omega \in \{a, b, c, d\}^* : \omega$ contiene a la palabra *abcd* como subcadena $\}$

   (b) $\{\omega \in \{a, b, c, d\}^* : \omega$ **NO** contiene a la palabra *abcd* como subcadena $\}$

   (c) $L_m = \{0^{n+m}1^n : n \geq 0\} \cup \{0^n 1^{n+m} : n \geq 0\}$

   (d) $\bigcup_{m=0}^{\infty} L_m$
   (Aquí $L_m$ es como en el problema anterior)

   (e) $\{0^{n^2} : n \geq 1\}$

   (f) $\{0^p : p$ es primo $\}$.

(g) {Sonetos en español}
(suponga que el conjunto de cadenas del "español" está bien definido)

(h) {Paréntesis balanceados} =
$= \{(), (()), ()(), ((())), (())(), ()(()), ()()(), (((()))), ((()())), \ldots, ((()))(), \ldots\}$

(i) Cadenas de paréntesis balanceados que no tengan más de un trillón de paréntesis.

(j) {Expresiones aritméticas} =
$= \{\omega \in \{+, -, *, /, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}^* :$ etcétera $\}$
(considerar a "$-$" y "$+$" únicamente como operadores binarios).

(k) {Expresiones aritméticas con paréntesis} =
$= \{\omega \in \{+, -, *, /, (, ), 0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}^* :$ etcétera $\}$

(l) {Declaraciones de variables en C}

(m) {Programas en C}

7. Pruebe que los lenguajes regulares son cerrados bajo: complemento, unión e intersección.

8. Probar que los lenguajes regulares son cerrados bajo concatenación y cerradura de Kleene.

9. Realizar las transformaciones indicadas usando los algoritmos vistos en clase.

   (a) Transformar los AFN's de la Figura 1 en AFD's

   (b) Transformar los AFD's de la Figura 2 en expresiones regulares.

   (c) Transformar las siguientes expresiones regulares en AFN's

      1. $(0+1)^*00(0+1)^*$
      2. $(0+\varepsilon)(1+10)^*$
      3. $(0+1)^*0(0+1)^7$

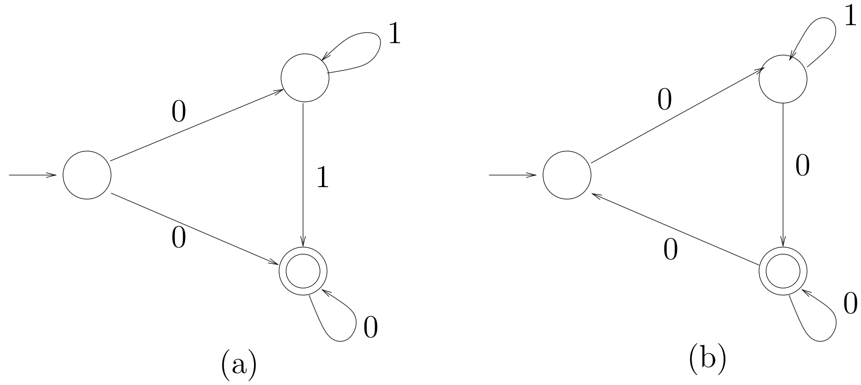


Figura 1: Convertir de AFN a AFD

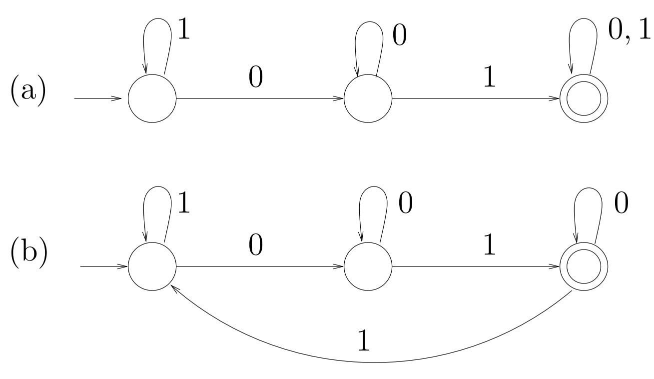


Figura 2: Convertir de AFD a expresión regular