Pauta de Corrección

Primer Certamen

Introducción a la Informática Teórica Informática Teórica

7 de mayo de 2016

1. Sean lenguajes $L_1 = \mathscr{L}(M_1)$ y $L_2 = \mathscr{L}(M_2)$ con DFAs $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ y $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$. Queremos definir un NFA que acepte el lenguaje $MIX(L_1, L_2)$. La idea es que siga el comportamiento de M_1 y M_2, alternando entre ellos al azar. El estado del NFA consta de los estados de M_1 y M_2, y el turno del que está en funciones, y vía transiciones ϵ cambia el turno. Formalmente:

   $$M = (Q_1 \times Q_2 \times \{1,2\}, \Sigma, \delta, (q_1, q_2, 1), F_1 \times F_2)$$

   Arbitrariamente comenzamos en el turno de M_1, puede cambiar al de M_2 vía ϵ. Definimos además:

   $$\delta((p_1, p_2, t), a) = \begin{cases} \{(\delta_1(p_1, a), p_2, 1)\} & t = 1 \\ \{(p_1, \delta_2(p_2, a), 2)\} & t = 2 \end{cases}$$
   $$\delta((p_1, p_2, t), \epsilon) = \{(p_1, p_2, 3 - t)\}$$

   O sea, podemos seguir el procesamiento del DFA de turno, o podemos cambiar espontáneamente el turno al otro. Aceptamos si ambos DFA aceptan.

   Una definición más simple es permitir que el NFA tome el paso de M_1 o M_2 directamente, sin considerar turnos:

   $$M = (Q_1 \times Q_2, \Sigma, \delta, (q_1, q_2), F_1 \times F_2)$$

   con:

   $$\delta((p_1, p_2), a) = \{(\delta_1(p_1, a), p_2), (p_1 \delta_2(p_2, a))\}$$

Puntajes

Total	25
Definición de los DFAs	4
Explicación del funcionamiento de autómata resultante	13
Descripción formal del autómata	8

2. Para demostrar que es de contexto libre basta da una gramática de contexto libre que lo genere:

$$
\begin{aligned}
S &\to aAa \\
A &\to bAb \mid a
\end{aligned}
$$

Para demostrar que no es regular, podemos usar directamente el lema de bombeo para lenguajes regulares. La demostración es por contradicción. Supongamos que L_2 es regular, por lo que cumple el lema de bombeo. Sea N la constante del lema, y consideremos la palabra $\sigma = ab^N ab^N a \in L_2$, donde $|\sigma| = 2N + 3 \geq N$. Por el lema, podemos escribir $\sigma = \alpha\beta\gamma$ tal que $|\alpha\beta| \leq N$ con $\beta \neq \epsilon$ y $\alpha\beta^k\gamma \in L_2$ para todo $k \geq 0$. Vemos que $\alpha\beta \in \mathscr{L}(ab^*)$, por lo que $\alpha \in \mathscr{L}(ab^*)$ (la a no puede ser parte de β, ya que de repetirse o dejarse fuera la palabra ya no tendría la forma $ab^* ab^* a$), y $\beta \in \mathscr{L}(b^+)$. Pero entonces:

$$
\alpha\beta^0\gamma = ab^{N-|\beta|} ab^N a \notin L_2
$$

El lenguaje L_2 no cumple el lema de bombeo, no es regular.

Otra forma es usar propiedades de clausura. Sea S la substitución:

$$
\begin{aligned}
L_a &= \{a\} \\
L_b &= \{c, d\}
\end{aligned}
$$

De esta forma (usando notación regular para simplificar):

$$
S(L_2) = \{a(c \mid d)^n a(c \mid d)^n a \colon n \geq 0\}
$$

Podemos segregar c y d:

$$
S(L_2) \cap \mathscr{L}(ac^+ ad^+ a) = \{ac^n ad^n a \colon n \geq 1\}
$$

Aplicamos el homomorfismo siguiente:

$$
\begin{aligned}
h(a) &= \epsilon \\
h(c) &= a \\
h(d) &= b
\end{aligned}
$$

y resulta:

$$
h(S(L_2) \cap \mathscr{L}(ac^* ad^* a)) = \{a^n b^n \colon n \geq 1\}
$$

Este es nuestro lenguaje no regular regalón. Si L_2 fuera regular, este lenguaje sería regular, llegamos a una contradicción.

Puntajes

Total		25
Demostrar que es de contexto libre	10	
Demostrar que no es regular	15	

3. Si L_3 fuera de contexto libre, lo sería el lenguaje

$$L_3 \cap \mathscr{L}(a^* b^* c^*) = \{a^n b^n c^n : n \geq 0\}$$

(intersección entre un lenguaje de contexto libre y uno regular), pero es nuestro ejemplo regalón de lenguaje que no es de contexto libre. Concluimos que L_3 no es de contexto libre.

Alternativamente, podemos aplicar directamente el lema de bombeo para lenguajes de contexto libre. Supongamos que L_3 es de contexto libre, sea N la constante del lema de bombeo. Consideremos la palabra $\sigma = a^N b^N c^N \in L_3$, con $|\sigma| = 3N \geq N$. Por el lema podemos escribir $\sigma = uvwxy$, con $|vwx| \leq N$, tal que $vx \neq \epsilon$ y para todo $k \geq 0$ tenemos $uv^k wx^k y \in L_3$. Vemos que vwx está formado a lo más por dos tipos de símbolo, eligiendo $k = 0$ deja de tener el mismo número de cada tipo de símbolo, y $uv^0 wx^0 y \neq L_3$. Esto contradice al lema, L_3 no es de contexto libre.

Puntajes

Total		25
Demostración	25	

4. Primero discutamos cómo puede construirse un PDA. La idea general es registrar mediante el número de A en el *stack* (respectivamente B) el exceso de a sobre b (respectivamente b sobre a). Mantenemos siempre Z_0 en la base del *stack* (de manera que no se nos trabe el PDA al tener el mismo número de cada tipo de símbolo en un momento cualquiera). Como lo que queremos hacer es contabilizar, es natural aceptar por *stack* vacío. Como c no interviene, tendremos movidas que simplemente dejan todo tal cual con este símbolo.

   De la discusión precedente queda claro que basta un único estado, nuestro PDA quedará descrito por:

   $$M = (\{q_0\}, \{a, b, c\}, \{Z_0, A, B\}, \delta, q_0, Z_0, \varnothing)$$

   y será $L_4 = \mathscr{N}(M)$.

   Dibujar el PDA dará un diagrama incomprensible, optaremos por otra descripción. Como hay un único estado, para determinar $\delta(q, x, X)$ basta indicar x y X, y el *string* por el que se reemplaza X en el *stack* en cada alternativa. Esto da el cuadro 1. En detalle, $\delta(q_0, \epsilon, Z_0)$ permite a M aceptar si el número

	$\boldsymbol{Z_0}$	$\boldsymbol{A}$	$\boldsymbol{B}$
$\boldsymbol{\epsilon}$	$\{\epsilon\}$	$\varnothing$	$\varnothing$
$\boldsymbol{a}$	$\{Z_0A\}$	$\{AA\}$	$\{\epsilon\}$
$\boldsymbol{b}$	$\{Z_0B\}$	$\{\epsilon\}$	$\{BB\}$
$\boldsymbol{c}$	$\{Z_0\}$	$\{A\}$	$\{B\}$

   Cuadro 1: Función de transición de M

   de a y b coinciden (no hay excedente de ninguno de los dos) y está al final de la palabra; las movidas $\delta(q_0, a, Z_0)$ y $\delta(q_0, a, A)$ aumentan el exceso de a sobre b, la movida $\delta(q_0, b, A)$ disminuye el exceso; las movidas para B en el *stack* son simétricas; un símbolo c en la entrada simplemente se ignora.

Puntajes

Total		25
Explicación del diseño	15	
Descripción (más o menos) formal del PDA	10	

5. Notamos que las palabras suficientemente largas son todas las descritas por $a^* b^*$. Formalmente:

$$L_5 = \{ab, a^2b^2, a^3b^3, a^4b^4\} \cup \mathscr{L}(aaa^* bbbbbb^*)$$

Ambos lenguajes son regulares, $\{ab, a^2b^2, a^3b^3, a^4b^4\}$ es finito y el otro está descrito por una expresión regular. Siendo L_5 la unión de lenguajes regulares, es regular.

Como L_5 es regular, es de contexto libre.

Puntajes

Total			25
Demostrar que L_5 es regular		20	
– "Suficientemente largos" están en $a^* b^*$	7		
– Descripción como unión de finito y regular	8		
– Aplicar clausura para concluir que es regular	5		
Como L_5 es regular, es de contexto libre		5	

6. Vemos que para todo $n \geq 0$:

$$E \Rightarrow^* ({}^n a)^n$$

(mediante $E \Rightarrow T \Rightarrow F \Rightarrow (E)$ obtenemos $({}^n E)^n$, luego $E \Rightarrow T \Rightarrow F \Rightarrow a$ entrega $({}^n a)^n$). Esto sugiere aplicar directamente el lema de bombeo para lenguajes regulares.

Supongamos que el lenguaje generado es regular, y sea N la constante del lema de bombeo. Consideremos la palabra $({}^N a)^N$, cuyo largo es $2N + 1 \geq N$. Vemos que en cualquier división en $\alpha\beta\gamma$ con $|\alpha\beta| \leq N$, la palabra β estará formada solo por (, repitiéndola (caso $k = 2$ del lema de bombeo) resulta $\alpha\beta^2\gamma$ que viola la regla sagrada de "todo lo que se abre se cierra". Esta palabra no pertenece al lenguaje, este no es regular.

Puntajes

Total		10
Constante del lema de bombeo	1	
Elegir palabra para bombear	3	
Subdivisión	2	
Palabra bombeada (elegir k) no pertenece al lenguaje	3	
Conclusión	1	