

## TEORÍA DE AUTÓMATAS Y LENGUAJES FORMALES

JUNIO 2009

### Final

---

NORMAS :

- La duración del examen es de 2 horas y 30 minutos.
- Todos los ejercicios se entregarán en hojas separadas.
- El examen tipo test cuenta hasta 2 puntos sobre la nota total.
- Además de obtener un mínimo de 5 puntos en el global del examen, para poder aprobar es necesario obtener un mínimo de 1.3 puntos entre las dos primeras preguntas y un mínimo de 1.3 puntos entre las dos últimas.

1. (a) (1.0 puntos) Define una gramática formal para el lenguaje:

$$L = \{axax | x \in \{a, b\}^*\}.$$

¿De que tipo de la jerarquía de Chomsky es la gramática que has definido?

**Solución:**

Una posible solución podría ser la siguiente:

$$G = (\{S, X, A, B, Y\}, \{a, b\}, S, P) \text{ con } P = \{S ::= aXaY; \quad X ::= XaA|XbB|\lambda; \\ Ba ::= aB; \quad Bb ::= bB; \quad Aa ::= aA; \quad Ab ::= bA; \\ AY ::= Ya; \quad BY ::= Yb; \quad aY ::= a\}$$

- (b) (0.6 puntos) Define una gramática regular, lineal por la derecha (según la definición dada en clase) para el lenguaje:

$$L = \{x \in \{a, b\}^* | \text{el segundo y el penultimo simbolo es una } a\}.$$

Considera que, por ejemplo, las palabras aa y baa pertenecen a  $L$ . Ten en cuenta que en una gramática regular tiene que haber exactamente un símbolo terminal en la parte derecha de cada producción y puede haber como mucho un símbolo no terminal (con la excepción de la regla  $S ::= \lambda$  siendo  $S$  el axioma).

**Solución:**

$$\text{Una posible solución es la siguiente: } G = (\{S\}, \{a, b\}, S, P) \text{ con } P = \{S ::= aB|bB|aE; \quad B ::= aC|aD; \quad C ::= aC|bC|aD; \quad D ::= a|b; \quad E ::= a\}$$

2. (1.6 puntos) Demuestra que el lenguaje el lenguaje generado por la gramática  $G_1$  es el mismo que el lenguaje reconocido por el autómata finito  $A_2$ . Todos los pasos para la resolución del ejercicio tienen que estar justificados utilizando algoritmos vistos en clase. No es obligatorio, pero sí conveniente, dar una pequeña explicación de qué se hace y porqué se hace.

$$G_1 = (\{S, B, C, D\}, \{a, b, c\}, S, P)$$

$$\begin{aligned}
 P &= \{S ::= aB \mid \lambda \\
 &\quad B ::= bC, \\
 &\quad C ::= cD \mid c \\
 &\quad D ::= S \mid B
 \end{aligned}$$

$$A_2 = (\{a, b, c\}, \{q_0, \dots, q_9\}, f_2, q_0, \{q_0, q_6, q_9\})$$

$f_2$	a	b	c
$\rightarrow^* q_0$	$q_1$	$q_2$	$q_3$
$q_1$	$q_2$	$q_4$	$q_3$
$q_2$	$q_3$	$q_3$	$q_5$
$q_3$	$q_3$	$q_3$	$q_3$
$q_4$	$q_3$	$q_3$	$q_6$
$q_5$	$q_3$	$q_3$	$q_5$
$^* q_6$	$q_7$	$q_8$	$q_3$
$q_7$	$q_2$	$q_4$	$q_5$
$q_8$	$q_3$	$q_3$	$q_9$
$^* q_9$	$q_7$	$q_8$	$q_5$

### Solución:

Como los dos elementos dados reconocen lenguajes regulares (uno es un AFND y el otro es una gramática regular), procederemos a comprobar los autómatas mínimos equivalentes de cada uno de ellos. Si los autómatas mínimos son isomorfos, entonces ambos elementos reconocen el mismo lenguaje.

Dada la gramática regular  $G_1$ , definimos el AFND correspondiente

$$A_1 = (\{a, b, c\}, \{S, B, C, D\}, f_1, S, \{S, D\})$$

$f_1$	a	b	c	$\lambda$
$\rightarrow^* S$	$\{B\}$			
$B$		$\{C\}$		
$C$			$\{D\}$	
$^* D$				$\{S, B\}$

Ahora, obtendremos un autómata finito determinista. Para ello, partimos de la  $clausura_\lambda(S) = \{S\}$ .

$f_1$	a	b	c
$\rightarrow^* \{S\}$	$\{B\}$	$\emptyset$	$\emptyset$
$\{B\}$	$\emptyset$	$\{C\}$	$\emptyset$
$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
$\{C\}$	$\emptyset$	$\emptyset$	$\{S, B, D\}$
$^* \{S, B, D\}$	$\{B\}$	$\{C\}$	$\emptyset$

Renombramos los estados

$f_1$	a	b	c
$\rightarrow^* q_0$	$q_1$	$q_2$	$q_2$
$q_1$	$q_2$	$q_3$	$q_2$
$q_2$	$q_2$	$q_2$	$q_2$
$q_3$	$q_2$	$q_2$	$q_4$
$^* q_4$	$q_1$	$q_3$	$q_2$

Vamos a minimizar el autómata para obtener el autómata finito determinista mínimo (y único).

$$Q/E_0 = \{\{q_0, q_4\}, \{q_1, q_2, q_3\}\}$$

$$Q/E_1 = \{\{q_0, q_4\}, \{q_1, q_2\}, \{q_3\}\}$$

$$Q/E_2 = \{\{q_0\}, \{q_4\}, \{q_1\}, \{q_2\}, \{q_3\}\}$$

$$Q/E_3 = \{\{q_0\}, \{q_4\}, \{q_1\}, \{q_2\}, \{q_3\}\} = Q/E$$

Como podemos comprobar, este autómata ya es mínimo.

Ahora, dado el AFD  $A_2$ , hallaremos el autómata mínimo equivalente:

$$Q/E_0 = \{\{q_0, q_6, q_9\}, \{q_1, q_2, q_3, q_4, q_5, q_7, q_8\}\}$$

$$Q/E_1 = \{\{q_0, q_6, q_9\}, \{q_1, q_2, q_3, q_5, q_7\}, \{q_4, q_8\}\}$$

$$Q/E_2 = \{\{q_0\}, \{q_6, q_9\}, \{q_1, q_7\}, \{q_2, q_3, q_5\}, \{q_4, q_8\}\}$$

$$Q/E_3 = \{\{q_0\}, \{q_6, q_9\}, \{q_1, q_7\}, \{q_2, q_3, q_5\}, \{q_4, q_8\}\} = Q/E$$

El autómata mínimo es:

$f_{det}$	a	b	c
$\rightarrow^* A$	$C$	$D$	$D$
$^* B$	$C$	$E$	$D$
$C$	$D$	$E$	$D$
$D$	$D$	$D$	$D$
$E$	$D$	$D$	$B$

Como podemos comprobar, los autómatas mínimos  $A_1$  y  $A_2$  son isomorfos. Por lo tanto, hemos demostrado que la gramática y el autómata finito dados en el enunciado definen el mismo lenguaje.

3. (1.6 puntos) Dado el lenguaje  $L = \{0^i 1^i 2^j \mid j = 2i, i, j \geq 0\}$ , demuestra, argumentando tu respuesta, si  $L$  es o no independiente del contexto.

**Solución:**

Supongamos que el lenguaje  $L$  es independiente del contexto. Entonces debe verificar el lema de bombeo para lenguajes independientes de contexto. Para la constante  $n$ , se elige la palabra  $z = 0^n 1^n 2^{2n}$ , cuya longitud es mayor a  $n$  y que pertenece a  $L$ . Se intenta descomponer la palabra elegida de la siguiente forma:  $z = u.v.w.x.y$ . Se analizarán todas las descomposiciones posibles:

1.  $vx$  contiene algún 0:  $vx$  no puede contener ningún 2. La palabra  $uv^2wx^2y$  contiene por lo menos  $(n+1)$  ceros y por lo menos  $n$  unos, pero sólo  $2n$  doses. Esta palabra no puede pertenecer a  $L$  ya que el número de doses es menor que la suma de ceros y unos.
2.  $vx$  contiene algún 1 (pero ningún 0 como demostrado anteriormente): La palabra  $uv^2wx^2y$  contiene por lo menos  $(n+1)$  unos. Esta nueva palabra no puede pertenecer a  $L$  ya que el número de ceros y unos no coincide.
3.  $vx$  contiene algún 2 (pero ningún 0 ni 1 como demostrado anteriormente): La palabra  $uv^2wx^2y$  contiene por lo menos  $(2n+1)$  doses. Esta palabra no puede pertenecer a  $L$  porque la suma de ceros y unos ( $2n$ ) es inferior.

En todos los casos llegamos a una contradicción, que parte de suponer que  $L$  es un lenguaje independiente del contexto. Concluimos que el lenguaje  $L$  no es un lenguaje independiente del contexto.

4. (0.8 + 0.8<sup>1</sup> puntos) Un grupo de físicos está capturando señales procedentes del espacio. Estos datos constan de pulsos en tres distintos rangos de frecuencias:  $a$ ,  $b$  y  $c$ .

Los científicos no están interesados en todos los datos que están recibiendo, únicamente les interesa una secuencia específica de pulsos:

1. La secuencia tiene que empezar con  $i$  señales del rango  $a$ .
2. Seguidamente tienen que llegar  $j$  señales del rango  $b$ .
3. El tercer grupo tiene que ser del rango  $c$ , pero la cantidad de señales tiene que coincidir con el número de señales del primer grupo  $i$  o con el número de señales del segundo grupo  $j$ .
4. Un cuarto, y último, grupo de señales tiene que proceder del rango  $a$  y su cantidad tiene que coincidir con el primer grupo de datos  $i$ .

¿Les puedes ayudar y construir un autómata a pila reconocedor por vaciado de pila que acepte esta secuencia de señales ( $a^i b^j c^k a^i$ , donde  $i, j > 0$ ,  $k = i$  o  $k = j$ )?

---

<sup>1</sup>0.8 puntos se obtienen si se cumple una de las dos condiciones del final del enunciado

Para mejorar el rendimiento de reconocimiento de las señales es necesaria una implementación en hardware. Para poder llevar a cabo esta tarea, el autómata a pila tiene que cumplir con una (no con las dos) de las siguientes condiciones:

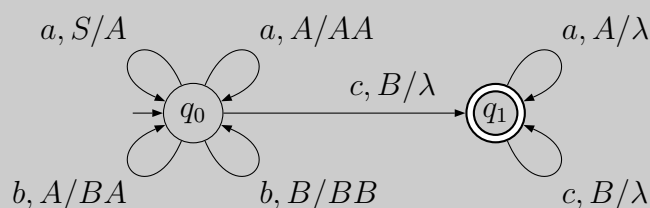
- La solución tiene que tener como mucho 5 estados.
- El alfabeto de pila de la solución tiene como máximo 4 distintos símbolos, es decir, el axioma más 3 símbolos adicionales.

### Solución:

Hay que explicar a los científicos que no se puede diseñar un autómata a pila que reconozca el lenguaje para el caso  $k = 1$ , ya que no representa un lenguaje independiente de contexto (demostrable mediante lema de bombeo para lenguajes independientes de contexto).

Para el otro caso  $k = j$ , sí se puede construir el siguiente autómata a pila:

$AP = (\{a, b, c\}, \{a, b, c, A, B, S\}, \{q_0, q_1\}, q_0, S, f)$ , donde  $f$  según el grafo



5. (1.6 puntos) Mi amigo Pepe Grillo inventó la siguiente forma normal de una gramática independiente del contexto, llamada FNPG (Forma normal de Pepe Grillo). Todas las producciones de la gramática FNPG tienen la siguiente forma:

$$A ::= abX, \text{ donde } A \in \Sigma_N, a, b \in \Sigma_T, X \in (\Sigma_N \cup \Sigma_T)^*$$

Es decir, todos los cuerpos de todas las producciones comienzan con dos símbolos terminales.

Observación: la única excepción es la producción  $S ::= \lambda$ , siendo  $S$  el axioma, que permite generar la palabra vacía en caso de que  $\lambda \in L$ .

Dada la siguiente gramática en FNPG  $G_{PG} = (\{S, A, B\}, \{a, b\}, S, P), :$

$$\begin{aligned} S &::= aaS \mid abS \mid baS \mid bbS \mid aaA \mid abA \mid baA \mid bbA \\ A &::= aa \mid ab \mid ba \mid bb \end{aligned}$$

- (a) (0.1 puntos) Demuestra mediante derivación que la palabra  $abaabbba$  pertenece al lenguaje  $L$  generado por la gramática  $G_{PG}$ .
- (b) (0.3 puntos) Demuestra mediante la aplicación del algoritmo CYK, que la palabra  $abba$  pertenece al lenguaje  $L$  generado por  $G_{PG}$ .

- (c) (0.2 puntos) Convierte la gramática  $G_{PG}$  en un autómatas a pila con un algoritmo visto en clase.
- (d) (0.1 puntos) Describe el lenguaje  $L_{PG}$  generado por  $G_{PG}$ .
- (e) (0.2 puntos) Pepe Grillo dice que puede generar cualquier gramática independiente del contexto con su forma normal FNPG. ¿Tiene razón? ¿Qué le dirías?
- (f) (0.7 puntos) Inventa un algoritmo que convierta una gramática en forma normal de Greibach (FNG) en una gramática en forma normal de Pepe Grillo (FNPG).

**Solución:**

(a)  $S \Rightarrow abA \Rightarrow abaaS \Rightarrow abaabbA \Rightarrow abaabbbA$

(b) Primer paso, convertir  $G_{PG}$  en una gramática en forma normal de Chomsky.

La gramática  $L_G$  ya está bien formada, por tanto, se convierte en FNC mediante los dos siguientes pasos:

1. Conversión de símbolos terminales:

$X ::= a; Y ::= b.$

$S ::= XXS \mid XYS \mid YXS \mid YYS;$

$S ::= XXA \mid XYA \mid YXA \mid YYA;$

$A ::= XX \mid XY \mid YX \mid YY;$

2. Conversión de las producciones:

$X ::= a; Y ::= b.$

$T_1 ::= XX; T_2 ::= XY; T_3 ::= YX; T_4 ::= YY;$

$S ::= T_1S \mid T_2S \mid T_3S \mid T_4S;$

$S ::= T_1A \mid T_2A \mid T_3A \mid T_4A;$

$A ::= XX \mid XY \mid YX \mid YY;$

Ahora, se puede aplicar el algoritmo CYK:

$X_{14} = \{S\}$			
$X_{13} = \{\}$	$X_{24} = \{\}$		
$X_{12} = \{T_2, A\}$	$X_{23} = \{T_4, A\}$	$X_{34} = \{T_3, A\}$	
$X_{11} = \{X\}$	$X_{22} = \{Y\}$	$X_{33} = \{Y\}$	$X_{44} = \{X\}$

Por lo tanto, la palabra  $abba$  pertenece la language  $L$  generado por  $G_{PG}$ .

(c) El autómatas a pila que reconoce por vaciado de pila tiene la siguiente descripción:

$A_V = (\{a, b\}, \{S, A, a, b\}, \{q\}, q, S, f_V)$ , donde  $f_V$

$f_V(q, \lambda, S) = \{(q, aaS), (q, abS), (q, baS), (q, bbS)\}$

$f_V(q, \lambda, S) = \{(q, aaA), (q, abA), (q, baA), (q, bbA)\}$

$f_V(q, \lambda, A) = \{(q, aa), (q, ab), (q, ba), (q, bb)\}$

$f_V(q, a, a) = \{(q, \lambda)\}$

$f_V(q, b, b) = \{(q, \lambda)\}$

- (d) El lenguaje  $L_{PG}$  consta de palabra sobre el alfabeto  $\{a, b\}$  que contiene palabras de longitud mayor a cuatro y par. Existen muchas maneras de describir este lenguaje, pero quizás la más intuitiva es la siguiente:

$$L_{PG} = \{ww^+ \mid w \in \{aa, ab, ba, bb\}^*\}.$$

- (e) Pepe Grillo se equivoca, no se puede representar TODOS los lenguajes independientes del contexto en FNPG. Aunque se pueda generar TODAS las palabras con una longitud mayor o igual a dos, no se puede generar NINGUNA palabra con una longitud igual a uno.
- (f) Como se ha visto en el ejercicio (e), es imposible generar palabras de longitud uno con la FNPG, por tanto, se omiten estas producciones. La única manera de generar estas palabras de longitud uno es desde el axioma, es decir, producciones del tipo  $A ::= a$ ,  $A \in \Sigma_N, a \in \Sigma_T$  solo existen para  $A$  siendo el axioma. Todas las demás producciones siguen la definición del enunciado.

Una gramática en forma normal de Greibach comienza siempre con un símbolo terminal, por tanto hay dos posibilidades de producciones que se diferencian en el segundo símbolo, que puede ser un terminal o una variable.

1.  $A ::= abX$ ,  $A \in \Sigma_N$ ,  $a, b \in \Sigma_T$ ,  $X \in (\Sigma_N \cup \Sigma_T)^*$
2.  $A ::= aBX$ ,  $A, B \in \Sigma_N$ ,  $a \in \Sigma_T$ ,  $X \in (\Sigma_N \cup \Sigma_T)^*$

Las producciones del tipo 1 ya se encuentran en forma normal de Pepe Grillo, por lo tanto, únicamente hace falta convertir las producciones del tipo 2.

En las producciones  $A ::= aBX$  podemos reemplazar el símbolo  $B$  por todos los cuerpos de sus producciones,

$$B ::= b_1Y_1 \mid b_2Y_2 \mid \dots \mid b_nY_n.$$

Observe, que todas las producciones tiene la forma normal de Greibach (FNG), es decir, que comienzan con un símbolo terminal.

Por tanto, obtenemos las siguientes producciones para  $A$ :

$$A ::= a_1 \underbrace{b_1Y_1}_B X \mid a_2 \underbrace{b_2Y_2}_B X \mid \dots \mid a_n \underbrace{b_nY_n}_B X$$

Todas estas producciones tienen la forma normal de Pepe Grillo (FNPG).