



UNIVERSIDADE
DE VIGO

Preguntas y respuestas para la evaluación continua **2008/2009**

Dr. Arno Formella
Universidad de Vigo
Departamento de Informática
Área de Lenguajes y Sistemas Informáticos
E-32004 Ourense

<http://www.ei.uvigo.es/~formella>

`formella@ei.uvigo.es`

9 de junio de 2009

Índice

1. Explicaciones	3
2. Hoja 1 (17 de Febrero de 2009)	4
3. Hoja 2 (03 de Marzo de 2009)	5
4. Hoja 3 (10 de Marzo de 2009)	7
5. Hoja 4 (17 de Marzo de 2009)	8
6. Hoja 5 (24 de Marzo de 2009)	9
7. Hoja 6 (31 de Marzo de 2009)	10
8. Hoja 7 (21 de Abril de 2009)	11
9. Hoja 8 (05 de Mayo de 2009)	13
10. Hoja 9 (12 de Mayo de 2009)	14
11. Hoja 10 (19 de Mayo de 2009)	15
12. Hoja 11 (26 de Mayo de 2009)	18

1. Explicaciones

- Habrá 12 hojas.
- Los nuevos problemas se distribuyen los martes.
- Las entregas se realizan de forma personal el siguiente martes con clases presenciales.
- Una entrega consiste en **una hoja** escrito **a mano** (es decir, ni impreso ni copia) personalmente.
- No hay excepciones para dicha forma de entrega.
- No se publican los resultados/correcciones individuales, sino se publica una posible solución correcta.
- El alumno debe autoevaluarse con las soluciones publicadas, la copia de la entrega guardada y mejor en grupo de compañeros.
- Tres de las doce entregas se seleccionarán al azar para su incorporación en el proceso de evaluación final de la asignatura (mira Guía Docente).

2. Hoja 1 (17 de Febrero de 2009)

P1: Enumera 3 situaciones en el contexto de la informática dónde crees el uso de lenguajes formales y sus autómatas correspondientes es “útil”.

R1:

- Verificación de la sintaxis correcta de cadenas de símbolos (p.ej.: direcciones de correo electrónico, numeros “reales” en programas, ficheros HTML o XML).
- Desarrollo de algoritmos (p.ej.: algoritmo para la búsqueda de una palabra (con o sin “comodines”) en un texto)
- Especificaciones de entradas/ficheros válidas.
- Comprobación si un problema es “computable” o no lo es.
- Verificación de sistemas basados en estados (p.ej.: un semáforo en un cruce).

P2: Comprueba/contradice: $\epsilon \in L^+$ si y solo si $\epsilon \in L$ (siendo L un lenguaje).

R2: Es verdad. Comprobamos (las dos direcciones):

1. Si $\epsilon \in L$ entonces $\epsilon \in L^+$ dado que $L = L^1 \subset L^+$ (según definición).
2. Si $\epsilon \in L^+$ entonces ϵ está en uno de los L^i ($i \geq 1$). Si $i = 1$ entonces $\epsilon \in L^1 = L$ y ya está. Asumimos $i > 1$, entonces sea i el índice más pequeño tal que $\epsilon \in L^i$. Con $L^i = L^{i-1}.L$ tiene que existir una palabra $w \in L^{i-1}$ diferente a ϵ (porque hemos dicho i es mínimo), a la cual añadimos algo para obtener ϵ . Eso es imposible, entonces i no puede ser mayor que 1.

P3: Comprueba/contradice: $L^+ = L^*$ si y solo si $\epsilon \in L$ (siendo L un lenguaje).

R3: Es verdad. Comprobamos (las dos direcciones):

1. Si $\epsilon \in L$ entonces $L^0 = \{\epsilon\} \subset L$, luego $L \subset L^+$ (por definición), con eso $L^0 \subset L^+$, entonces $L^* = L^+ \cup L^0 = L^+$ (el primer = por definición).
2. Si $L^+ = L^*$, entonces $\epsilon \in L^+$ dado que $\epsilon \in L^*$ (recordar $\{\epsilon\} = L^0 \subset L^*$ según definición). Entonces, según **R2**, $\epsilon \in L$.

3. Hoja 2 (03 de Marzo de 2009)

P1: Sea $N = \{0, 1, 2, 3, 4, 5, 6, 7\}$ un conjunto (de dígitos), y sea

$$R = \{(0, 3), (1, 4), (2, 0), (3, 1), (4, 2), (5, 6), (6, 7)\} \subset N \times N$$

una relación sobre N .

- Construye las relaciones R^i para todos los $i = 0, \dots, \infty$.
- Construye la relación R^* .
- Argumenta si R^* es reflexiva, simétrica, y/o transitiva.
- ¿Cuál pareja deberíamos añadir a la relación R para que R^* sea simétrica (si piensas que R^* no es simétrica en el apartado anterior).

R1:

- Los conjuntos R^i son:

$$R^0 = \{(0, 0), (1, 1), (2, 2), (3, 3), (4, 4), (5, 5), (6, 6), (7, 7)\} \quad (1)$$

$$R^1 = \{(0, 3), (1, 4), (2, 0), (3, 1), (4, 2), (5, 6), (6, 7)\} \quad (2)$$

$$R^2 = \{(0, 1), (1, 2), (2, 3), (3, 4), (4, 0), (5, 7)\} \quad (3)$$

$$R^3 = \{(0, 4), (1, 0), (2, 1), (3, 2), (4, 3)\} \quad (4)$$

$$R^4 = \{(0, 2), (1, 3), (2, 4), (3, 0), (4, 1)\} \quad (5)$$

$$R^5 = \{(0, 0), (1, 1), (2, 2), (3, 3), (4, 4)\} \quad (6)$$

$$R^6 = \{(0, 3), (1, 4), (2, 0), (3, 1), (4, 2)\} \quad (7)$$

$$R^7 = \{(0, 1), (1, 2), (2, 3), (3, 4), (4, 0)\} \quad (8)$$

para $n > 7$

$$R^8 = R^3 \quad (9)$$

$$R^9 = R^4 \quad (10)$$

$$R^{10} = R^5 \quad (11)$$

$$R^{11} = R^6 \quad (12)$$

$$R^{12} = R^7 \quad (13)$$

es decir

$$R^n = R^{(n-3) \bmod 5+3} \quad (14)$$

- El conjunto R^* es (basta con unir hasta R^7):

$$\begin{aligned}
 R^* = \{ & (0, 0), (0, 1), (0, 2), (0, 3), (0, 4), \\
 & (1, 0), (1, 1), (1, 2), (1, 3), (1, 4), \\
 & (2, 0), (2, 1), (2, 2), (2, 3), (2, 4), \\
 & (3, 0), (3, 1), (3, 2), (3, 3), (3, 4), \\
 & (4, 0), (4, 1), (4, 2), (4, 3), (4, 4), \\
 & (5, 5), (5, 5), (5, 7), (6, 6), (6, 7), (7, 7) \}
 \end{aligned} \tag{15}$$

- **reflexiva:** sí, dado que $R^0 \subset R^*$
simétrica: no, dado que $(5, 7) \in R^*$ pero $(7, 5) \notin R^*$
transitiva: sí, por construcción.
- Se añade $(7, 5)$ y se verifica fácilmente que

$$\begin{aligned}
 R^* = \{ & (0, 0), (0, 1), (0, 2), (0, 3), (0, 4), \\
 & (1, 0), (1, 1), (1, 2), (1, 3), (1, 4), \\
 & (2, 0), (2, 1), (2, 2), (2, 3), (2, 4), \\
 & (3, 0), (3, 1), (3, 2), (3, 3), (3, 4), \\
 & (4, 0), (4, 1), (4, 2), (4, 3), (4, 4), \\
 & (5, 5), (5, 6), (5, 7), \\
 & (6, 5), (6, 6), (6, 7), \\
 & (7, 5), (7, 6), (7, 7) \}
 \end{aligned} \tag{16}$$

4. Hoja 3 (10 de Marzo de 2009)

P1: Construye una gramática G que genera el lenguaje

$$L = \{a^i b^j c^k d^l \mid (i = j \text{ y } k = l) \text{ ó } (i = l \text{ y } j = k); i, j, k, l \geq 0\}$$

ejemplos de palabra que pertenecen a L :

$$aabbccddd, abcd, aaabbccddd, aaaabbbb, cccddd$$

R1: una posible solución

$$\begin{aligned} G = & (\{\$, A, B, C, D\}, \{a, b, c, d\}, \\ & \{\$ \longrightarrow \epsilon \mid AB \mid aCd \mid bDc, \\ & A \longrightarrow \epsilon \mid aAb, B \longrightarrow \epsilon \mid cBd, \\ & C \longrightarrow \epsilon \mid aCd \mid bDc, D \longrightarrow \epsilon \mid bDc\}, \$) \end{aligned}$$

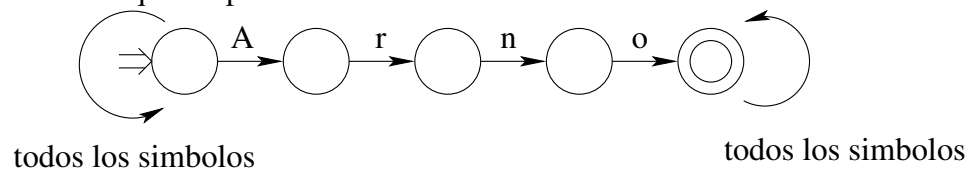
5. Hoja 4 (17 de Marzo de 2009)

P1: Si un autómata acepta la palabra ϵ , entonces el estado inicial es un estado final. ¿Es verdad?

R1: Depende: para los AFD y AFND, vistos hasta ahora en clase, es verdad; si permitimos transiciones ϵ no lo es. Miramos los AFD y los AFND: obviamente si el estado inicial es un estado final, tal autómata acepta ϵ , y al revés, si el estado inicial no es un estado final, entonces no acepta ϵ , dado que $\delta^*(q_0, \epsilon) = q_0$.

P2: Construye un autómata finito que acepta un texto (que no es nada más que una palabra larga sobre algún alfabeto) si en él aparece tu nombre.

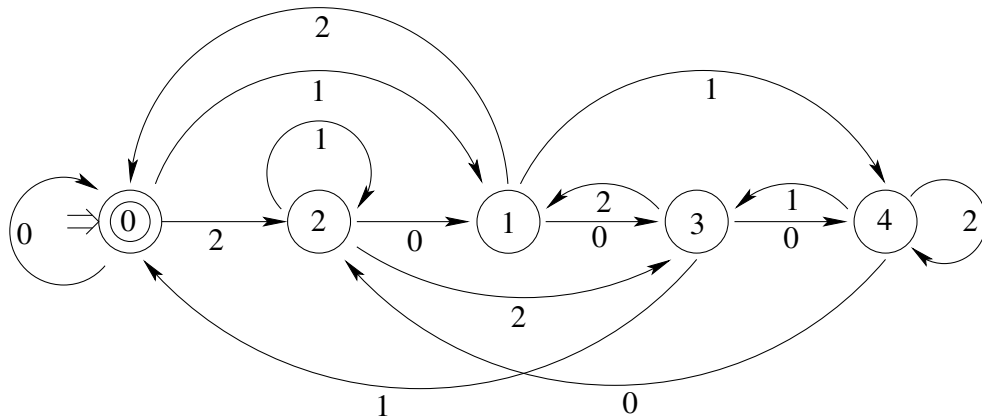
R2: Asumimos un alfabeto adecuado, p.ej. los caracteres imprimibles de la tabla del código ASCII. Un AFND que acepta el nombre “Arno” sería:



6. Hoja 5 (24 de Marzo de 2009)

P1: Construye una autómatas finito determinista (AFD) que acepta todas las palabras w sobre el alfabeto $\{0, 1, 2\}$ que son divisibles por 5 si se interpreta w como un número en la base 3.

R1:



7. Hoja 6 (31 de Marzo de 2009)

P1:

Convierte el autómata del examen de TALF de Junio 2008 en su pregunta 1 en un autómata finito determinista. Incluye en tu solución las tablas tal como lo vimos en clase.

(Sugerencia: aprovecha de la Semana Santa para realizar más ejercicios de antiguos exámenes.)

R1:

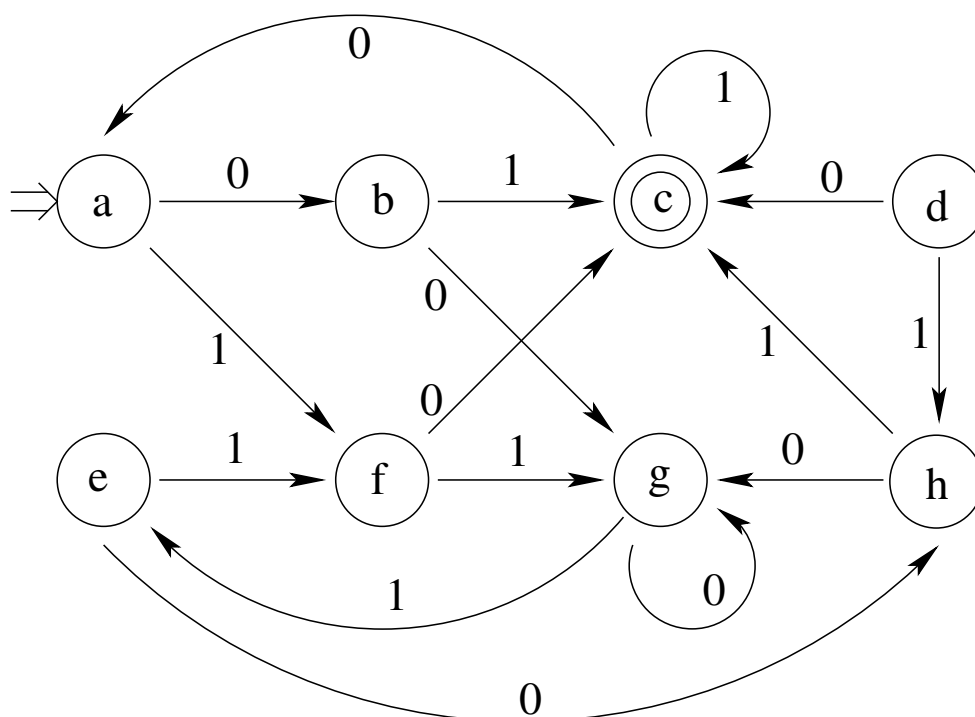
		a	b	ϵ	$cl-\epsilon$	a	b	a	b
$\longrightarrow$	0	2	-	1	0,1	2,4	1	2,4,5	1
*	1	4	1	-	1	4	1	2,4,5	1
	2	-	3	5	2,5	-	3,4	-	2,3,4,5
	3	4,5	-	-	3	4,5	-	2,4,5	-
	4	5	-	2	2,4,5	5	3,4	5	2,3,4,5
*	5	-	4	-	5	-	4	-	2,4,5

Dado que 1 (siendo estado final) está en la clausura- ϵ del estado inicial, el estado inicial también se convierte en estado final.

		a	b
$\longrightarrow$	* 0	2,4,5	1
	* 2,4,5	5	2,3,4,5
	* 1	2,4,5	1
	* 5	$\emptyset$	2,4,5
	* 2,3,4,5	2,4,5	2,3,4,5
	$\emptyset$	$\emptyset$	$\emptyset$

8. Hoja 7 (21 de Abril de 2009)

P1: Minimiza el siguiente autómata finito determinista.



El estado d no es alcanzable, se elimina. Se observa que el autómata está completo.

	a	b	c	e	f	g	h
a	-	X_1	X_0		X_2	X_6	X_3
b	-	-	X_0	X_4	X_5	X_6	
c	-	-	-	X_0	X_0	X_0	X_0
e	-	-	-	-	X_7	X_8	X_9
f	-	-	-	-	-	X_{10}	X_{11}
g	-	-	-	-	-	-	X_{12}
h	-	-	-	-	-	-	-

pareja	0	1	acción
(a, b)	(b, g)	(f, c)	X_1
(a, e)	(b, h)	(f, f)	(b, h)
(a, f)	(b, c)	(f, g)	X_2
(a, g)	(b, g)	(f, e)	$(b, g), (f, e)$
(a, h)	(b, g)	(f, c)	X_3
(b, e)	(g, h)	(c, f)	X_4
(b, f)	(g, c)	(c, g)	X_5
(b, g)	(g, g)	(c, e)	$X_6, (a, g)$
(b, h)	(g, g)	(c, c)	—
(e, f)	(h, c)	(f, g)	X_7
(e, g)	(h, g)	(f, e)	X_8
(e, h)	(h, g)	(f, c)	X_9
(f, g)	(c, g)	(g, e)	X_{10}
(f, h)	(c, g)	(g, c)	X_{12}

Entonces las parejas (a, e) y (b, h) marcan estados equivalentes.

P2: Minimiza el autómata finito determista de la hoja anterior (Hoja 6).

Renombrando los estados se obtiene la siguiente tabla de un AFD completo y conexo:

		a	b
→ *	0	1	2
*	1	3	4
*	2	1	2
*	3	5	1
*	4	1	4
	5	5	5

	0	1	2	3	4	5
0	-	X_2		X_1		X_0
1	-	-	X_2	X_2	X_3	X_0
2	-	-	-	X_4		X_0
3	-	-	-	-	X_5	X_0
4	-	-	-	-	-	X_0
5	-	-	-	-	-	-

pareja	a	b	acción
(0, 1)	(1, 3)	(2, 4)	(1, 3), (2, 4)
(0, 2)	(1, 1)	(2, 2)	—
(0, 3)	(1, 5)	(2, 1)	X_1
(0, 4)	(1, 1)	(2, 4)	(2, 4)
(1, 2)	(3, 1)	(4, 2)	(1, 3), (2, 4)
(1, 3)	(3, 5)	(4, 1)	$X_2, (0, 1), (1, 2)$
(1, 4)	(3, 1)	(4, 4)	X_3
(2, 3)	(1, 5)	(2, 1)	X_4
(2, 4)	(1, 1)	(2, 4)	—
(3, 4)	(5, 1)	(1, 4)	X_5

Entonces las parejas (0, 2), (2, 4), y (0, 4) marcan estados equivalentes, es decir, los tres entre si. Y el AFD mínimo es:

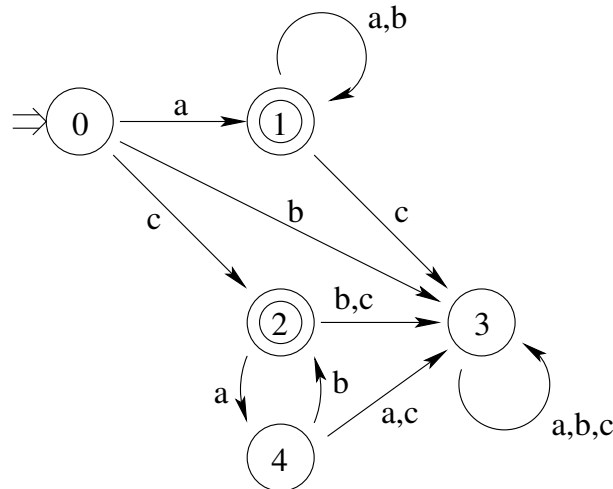
		a	b
→ *	0	1	0
*	1	3	0
*	3	5	1
	5	5	5

9. Hoja 8 (05 de Mayo de 2009)

P1: Construye un AFD mínimo para la siguiente expresión regular: $(a(ba)^*(a+b)^* + c(ab)^*)$ que define un lenguaje sobre $\{a, b, c\}$.

Hay diferentes formas de contestar:

1. Se construye un AFND- ϵ sucesivamente según visto en la comprobación que EE.RR. definen lenguajes regulares. Dicho autómata *grande* se convierte en AFND, luego en AFD, (luego en AFD completo sino ya está), y se minimiza.
2. Se construye un AFND- ϵ sucesivamente según visto en la comprobación que EE.RR. definen lenguajes regulares. Antes de la conversión a AFND, se unifica estados que obviamente son equivalentes. Así el AFND- ϵ se reduce considerablemente.
3. Se observa: $(ba)^*(a+b)^* \equiv (a+b)^*$. Con eso: $(a(ba)^*(a+b)^* + c(ab)^*) \equiv (a(a+b)^* + c(ab)^*)$. Y luego se puede construir directamente (en este caso fácil) un AFD completo



del cual se notará que ya es mínimo.

10. Hoja 9 (12 de Mayo de 2009)

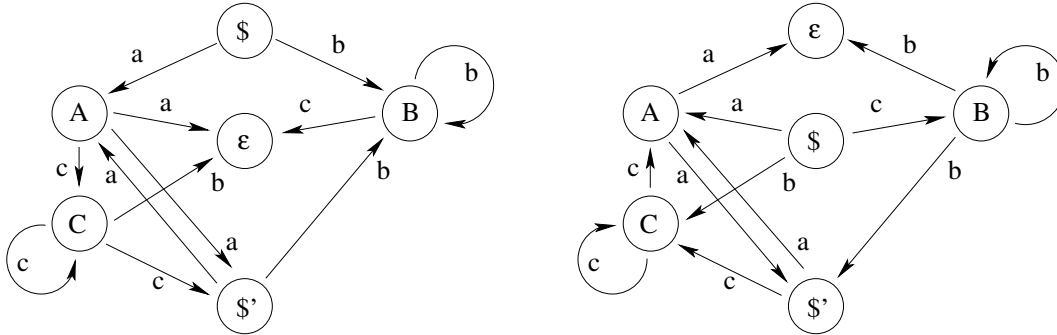
P1: Transforma la siguiente gramática lineal por la derecha en una gramática lineal por la izquierda:

$$G = (\{ \$, A, B, C \}, \{ a, b, c \}, \\ \{ \$ \rightarrow aA \mid bB, A \rightarrow a\$ \mid cC \mid a, B \rightarrow bB \mid c, C \rightarrow cC \mid c\$ \mid b \}, \$)$$

Sustituimos primero \$ a la derecha:

$$G = (\{ \$, \$', A, B, C \}, \{ a, b, c \}, \\ \{ \$ \rightarrow aA \mid bB, \$' \rightarrow aA \mid bB, \\ A \rightarrow a\$' \mid cC \mid a, B \rightarrow bB \mid c, C \rightarrow cC \mid c\$' \mid b \}, \$)$$

Generamos los grafos:



Leemos la gramática:

$$G = (\{ \$, \$', A, B, C \}, \{ a, b, c \}, \\ \{ \$ \rightarrow Aa \mid Bc \mid Cb, \$' \rightarrow Cc \mid Aa, \\ A \rightarrow \$'a \mid a, B \rightarrow Bb \mid b \mid \$'b, C \rightarrow Cc \mid Ac \}, \$)$$

11. Hoja 10 (19 de Mayo de 2009)

P1: Transforma la siguiente gramática libre de contexto en forma normal de Chomsky:

$$\begin{aligned}
 G = (\quad & \{ \$, A, B, C, D, E, F \}, \{ a, b, c \}, \\
 & \{ \$ \longrightarrow aAb \mid bB \mid DD, \\
 & A \longrightarrow a\$ \mid cC \mid a, \\
 & B \longrightarrow bBB \mid c \mid A, \\
 & C \longrightarrow cC \mid c\$ \mid b \mid DDCa, \\
 & D \longrightarrow \epsilon \mid Eb, \\
 & E \longrightarrow Eba \mid Fa, \\
 & F \longrightarrow AEb \} \\
 & , \$)
 \end{aligned}$$

Paso 1: Eliminar símbolos inútiles: iteramos sobre el conjunto de variables para encontrar las variables generativas. Las detectamos en el siguiente orden: $A, B, C, D, \$$. Entonces $N = \{E, F\}$ es el conjunto de variables no-generativas que eliminamos, queda P'_1 :

$$\begin{aligned}
 \$ & \longrightarrow aAb \mid bB \mid DD \\
 A & \longrightarrow a\$ \mid cC \mid a \\
 B & \longrightarrow bBB \mid c \mid A \\
 C & \longrightarrow cC \mid c\$ \mid b \mid DDCa \\
 D & \longrightarrow \epsilon
 \end{aligned}$$

Todas las restantes variables son accesibles.

Paso 2: Sustituir las constantes para obtener P'_2 :

$$\begin{aligned}
 \$ & \longrightarrow W_aAW_b \mid W_bB \mid DD \\
 A & \longrightarrow W_a\$ \mid W_cC \mid W_a \\
 B & \longrightarrow W_bBB \mid W_c \mid A \\
 C & \longrightarrow W_cC \mid W_c\$ \mid W_b \mid DDCW_a \\
 D & \longrightarrow \epsilon
 \end{aligned}$$

$$\begin{aligned}
W_a &\longrightarrow a \\
W_b &\longrightarrow b \\
W_c &\longrightarrow c
\end{aligned}$$

Paso 3: Confinar a longitud dos para obtener P'_3 :

$$\begin{aligned}
\$ &\longrightarrow W_a\$_1 \mid W_bB \mid DD \\
\$_1 &\longrightarrow AW_b \\
A &\longrightarrow W_a\$ \mid W_cC \mid W_a \\
B &\longrightarrow W_bB_1 \mid W_c \mid A \\
B_1 &\longrightarrow BB \\
C &\longrightarrow W_cC \mid W_c\$ \mid W_b \mid C_1C_2 \\
C_1 &\longrightarrow DD \\
C_2 &\longrightarrow CW_a \\
D &\longrightarrow \epsilon \\
W_a &\longrightarrow a \\
W_b &\longrightarrow b \\
W_c &\longrightarrow c
\end{aligned}$$

Paso 4: Eliminar producciones nulas: iteramos sobre las variables para detectar aquellas que posiblemente producen ϵ . Las detectamos en el orden reflejado en el conjunto: $E = \{D, \$, C_1\}$ de las cuales D y C_1 solamente generan ϵ , entonces $E_\epsilon = \{D, C_1\}$. Notamos que $\$ \in E$, por eso habrá paso 6. Obtenemos P'_4 :

$$\begin{aligned}
\$ &\longrightarrow W_a\$_1 \mid W_bB \\
\$_1 &\longrightarrow AW_b \\
A &\longrightarrow W_a\$ \mid W_cC \mid W_a \\
B &\longrightarrow W_bB_1 \mid W_c \mid A \\
B_1 &\longrightarrow BB \\
C &\longrightarrow W_cC \mid W_c\$ \mid W_b \mid W_c \mid C_2 \\
C_2 &\longrightarrow CW_a \\
W_a &\longrightarrow a \\
W_b &\longrightarrow b \\
W_c &\longrightarrow c
\end{aligned}$$

Paso 5: Eliminar producciones unitarias: calculamos la clausura:

$$\begin{aligned} U_1 &= \{(A, W_a), (B, W_c), (B, A), (C, W_b), (C, W_c), (C, C_2)\} \\ U_2 &= \{(B, W_a)\} \\ U_3 &= \{\} \end{aligned}$$

Obtenemos P'_5 (observa que hay que eliminar la variable C_2 que ya no es accesible):

$$\begin{aligned} \$ &\longrightarrow W_a\$_1 \mid W_bB \\ \$ _1 &\longrightarrow AW_b \\ A &\longrightarrow W_a\$ \mid W_cC \mid a \\ B &\longrightarrow W_bB_1 \mid c \mid W_a\$ \mid W_cC \mid a \\ B_1 &\longrightarrow BB \\ C &\longrightarrow W_cC \mid W_c\$ \mid b \mid c \mid CW_a \\ W_a &\longrightarrow a \\ W_b &\longrightarrow b \\ W_c &\longrightarrow c \end{aligned}$$

Paso 6: Permitir que se genera ϵ :

$$\begin{aligned} \$ &\longrightarrow W_a\$_1 \mid W_bB \mid \epsilon \\ \$' &\longrightarrow W_a\$_1 \mid W_bB \\ \$ _1 &\longrightarrow AW_b \\ A &\longrightarrow W_a\$' \mid W_cC \mid a \\ B &\longrightarrow W_bB_1 \mid c \mid W_a\$' \mid W_cC \mid a \\ B_1 &\longrightarrow BB \\ C &\longrightarrow W_cC \mid W_c\$' \mid b \mid c \mid CW_a \\ W_a &\longrightarrow a \\ W_b &\longrightarrow b \\ W_c &\longrightarrow c \end{aligned}$$

La gramática final es (siendo P el sistema de producciones anterior):

$$G = (\{\$, \$', A, B, B_1, C, W_a, W_b, W_c\}, \{a, b, c\}, P, \$)$$

12. Hoja 11 (26 de Mayo de 2009)

P1: Dado la siguiente gramática en forma normal de Greibach:

$$\begin{aligned}
 G = (& \{ \$, A, B, C, D, E \}, \{ a, b, c, d \}, \\
 & \{ \$ \longrightarrow \epsilon \mid aABB \mid aABBC \mid aB \mid aBC \mid cDEE \mid cE, \\
 & A \longrightarrow aAB \mid a, \\
 & B \longrightarrow b, \\
 & C \longrightarrow cDEE \mid cE, \\
 & D \longrightarrow cDE \mid c, \\
 & E \longrightarrow d \} \\
 & , \$)
 \end{aligned}$$

1. Construye un autómata finito con pila M que acepta el lenguaje generado por G , es decir, $L(M) = L(G)$.
2. Realiza un cálculo del autómata que acepta la palabra $aaabbbccdd$ y argumenta por qué el autómata no acepta abc .
3. ¿Qué lenguaje genera G ?

1. El autómata finito con pila es:

$$\begin{aligned}
 M = & (\{ a, b, c, d \}, \{ \$, A, B, C, D, E \}, \{ q \}, \delta, q, \$, \emptyset) \\
 \delta : & \delta(q, \epsilon, \$) = \{(q, \epsilon)\} \\
 & \delta(q, a, \$) = \{(q, ABB), (q, ABBC), (q, B), (q, BC)\} \\
 & \delta(q, c, \$) = \{(q, DEE), (q, E)\} \\
 & \delta(q, a, A) = \{(q, AB), (q, \epsilon)\} \\
 & \delta(q, b, B) = \{(q, \epsilon)\} \\
 & \delta(q, c, C) = \{(q, DEE), (q, E)\} \\
 & \delta(q, c, D) = \{(q, DE), (q, \epsilon)\} \\
 & \delta(q, d, E) = \{(q, \epsilon)\}
 \end{aligned}$$

- 2.

$$\begin{aligned}
 (q, aaabbbccdd, \$) & \vdash (q, aabbbccdd, ABBC) \\
 & \vdash (q, abbbccdd, ABBC) \\
 & \vdash (q, bbbccdd, BBBC)
 \end{aligned}$$

$$\begin{aligned}
&\vdash (q, bbccdd, BBC) \\
&\vdash (q, bccdd, BC) \\
&\vdash (q, ccdd, C) \\
&\vdash (q, cdd, DEE) \\
&\vdash (q, dd, EE) \\
&\vdash (q, d, E) \\
&\vdash (q, \epsilon, \epsilon)
\end{aligned}$$

Observamos porque no existe cálculo para abc :

- La producción $\$ \rightarrow \epsilon$ no genera abc , obviamente.
- Si generamos una vez una B , tal variable tenemos que sustituir por una b ; entonces $\$ \rightarrow aABB$ y $\$ \rightarrow aABBC$ nos producen dos b 's, y $\$ \rightarrow aB$ como mucho ab .
- Si generamos una vez una E , tal variable tenemos que sustituir por una d ; entonces las producciones $\$ \rightarrow cDEE$ y $\$ \rightarrow cE$ nos generan por lo menos una d , y $\$ \rightarrow aBC$ nos genera via C por lo menos una d .
- Es decir, empezamos como queramos desde $\$$, siempre llegamos a una contradicción.

3.
$$L(G) = \{a^i b^i c^j d^j \mid i, j \geq 0\}$$