Universidade de Vigo

Preguntas y respuestas
para la evaluación continua de
TALF
2009/2010

Dr. Arno Formella
Universidade de Vigo
Escola Superior de Enxeñaría Informática
Departamento de Informática
Área de Linguaxes e Sistemas Informáticos
E-32004 Ourense

http://www.ei.uvigo.es/~formella
formella@ei.uvigo.es

Abril 2010

Índice

1. Explicaciones

- Habrá unas 13 hojas.
- Los nuevos problemas se distribuen 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.
- Dicha hoja lleva en la parte derecha arriba los apellidos, el nombre y el DNI. (**¡Hojas que no cumplen con eso no serán ni mirados desde lejos!**)
- 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 entregas se seleccionarán al azar para su incorporación en el proceso de evaluación final de la asignatura (mira Guía Docente).

2. Hoja 1 (23 de Febrero de 2010)

P1: Busca y describe el significado de la siguiente secuencia de *símbolos*. Describe también como lo has conseguido.

正方形

R1: Quadrado.

P2: Enumera y comenta brevemente 3 situaciones en el contexto de la informática dónde el uso de lenguajes formales y sus autómatas correspondientes es "útil".

R2:

- Verificación de la sintaxis correcta de cadenas de símbolos (p.ej.: direcciones de correo electrónico, números "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). Ojo, aquí hay que trabajar con palabras de longitud posiblemente infinita.
- "Compilación" de programas.
- Diagramas de estados en lenguajes de modelado (por ejemplo UML).
- y mucho, mucho más...

3. Hoja 2 (2 de Marzo de 2010)

P1: Dado dos lenguajes L_1 y L_2 sobre el alfabeto $\Sigma = \{a, b, c\}$. Anotamos con $L_1 \cup L_2$ la unión, con $L_1 \cap L_2$ la intersección, con $\overline{L}_1$ el complemento, y con $L_1 - L_2$ la diferencia. Verifica o contradice:

- $\overline{L_1} = \Sigma^* - L_1$
- $\overline{L_1 \cup L_2} = \overline{L}_1 \cap \overline{L}_2$
- $\overline{L_1 \cap L_2} = \overline{L}_1 \cup \overline{L}_2$
- $L_1.L_2 = L_2.L_1$

R1: Si L_1 y L_2 son lenguajes sobre un alfabeto Σ, entonces $L_1 \subset \Sigma^*$ y $L_2 \subset \Sigma^*$ (por definición). Además sabemos por definición:

- $L_1 \cup L_2 = \{w \mid w \in L_1 \text{ y } w \in L_2\} \subset \Sigma^*$
- $L_1 \cap L_2 = \{w \mid w \in L_1 \text{ o } w \in L_2\} \subset \Sigma^*$
- $\overline{L}_1 = \{w \mid w \notin L_1\} \subset \Sigma^*$
- $L_1 - L_2 = \{w \mid w \in L_1 \text{ y } w \notin L_2\} \subset \Sigma^*$

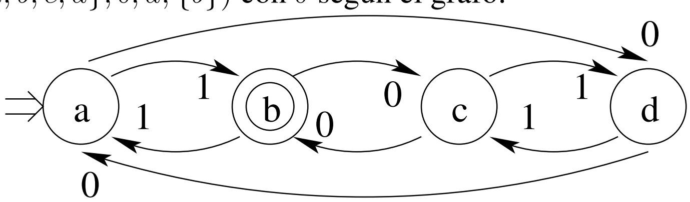
Entonces si usamos la definición de la diferencia con L_1 como Σ^* y L_2 como L_1, tenemos $\Sigma^* - L_1 = \{w \mid w \in \Sigma^* \text{ y } w \notin L_1\} = \{w \mid w \notin L_1\} = \overline{L}_1$. Usamos una tabla de pertenencia para alguna palabra w para comprobar la segunda y tercera equación (Leyes de De Morgan):

L_1	L_2	Σ^*	$\overline{L}_1$	$\overline{L}_2$	$L_1 \cup L_2$	$L_1 \cap L_2$	$\overline{L_1 \cup L_2}$	$\overline{L_1 \cap L_2}$	$\overline{L}_1 \cap \overline{L}_2$	$\overline{L}_1 \cup \overline{L}_2$
$\in$	$\in$	$\in$	$\notin$	$\notin$	$\in$	$\in$	$\notin$	$\notin$	$\notin$	$\notin$
$\in$	$\notin$	$\in$	$\notin$	$\in$	$\in$	$\notin$	$\notin$	$\in$	$\notin$	$\in$
$\notin$	$\in$	$\in$	$\in$	$\notin$	$\in$	$\notin$	$\notin$	$\in$	$\notin$	$\in$
$\notin$	$\notin$	$\in$	$\in$	$\in$	$\notin$	$\notin$	$\in$	$\in$	$\in$	$\in$

Y vemos que las columnas correspondientes son iguales. Para la última construimos un contraejemplo: Sea $L_1 = \{a\}$ y $L_2 = \{b\}$, entonces $L_1.L_2 = \{ab\}$ y $L_2.L_1 = \{ba\}$ y dichos conjuntos no son iguales, entonces la equación es incorrecta.

P2: Construye un autómata finito determinista que "acepta" el lenguaje L que contiene todas las palabras (finitas) sobre el alfabeto $\{0, 1\}$ con un número par de 0s y un número impar de 1s.

R2: $M = (\{0, 1\}, \{a, b, c, d\}, \delta, a, \{b\})$ con δ según el grafo:



4. Hoja 3 (9 de Marzo de 2010)

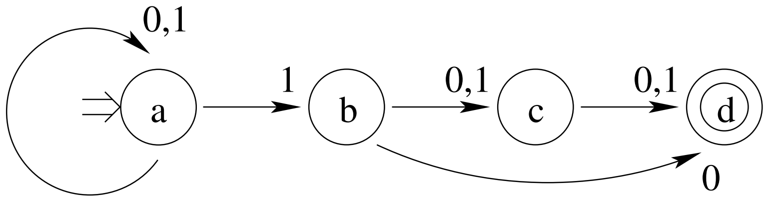
P1: Calcula, paso a paso, el resultado de la función δ^* para el autómata de la hoja anterior y la palabra de entrada $w = 0100110$, es decir, $\delta^*(a, w)$.

R1:

$$\begin{aligned}
\delta^*(a, 0100110) &= \delta^*(\delta(a, 0), 100110) = \delta^*(d, 100110)\\
&= \delta^*(\delta(d, 1), 00110) = \delta^*(c, 00110)\\
&= \delta^*(\delta(c, 0), 0110) = \delta^*(b, 0110)\\
&= \delta^*(\delta(b, 0), 110) = \delta^*(c, 110)\\
&= \delta^*(\delta(c, 1), 10) = \delta^*(d, 10)\\
&= \delta^*(\delta(d, 1), 0) = \delta^*(c, 0)\\
&= \delta^*(\delta(c, 0), \epsilon) = \delta^*(b, \epsilon) = b
\end{aligned}$$

es decir, el AFD acepta la palabra, dado que b es un estado final del AFD.

P2: Escribe la tabla para la función de transición δ del AFND con el siguiente grafo:



Calcula, paso a paso, el resultado de la función δ^* para este autómata y la palabra de entrada $w = 0100110$, es decir, $\delta^*(a, w)$. Averigua una palabra sobre $\{0, 1\}$ que el autómata no acepta.

R2:

δ	0	1
$\Longrightarrow a$	$\{a\}$	$\{a, b\}$
b	$\{c, d\}$	$\{c\}$
c	$\{d\}$	$\{d\}$
$\star\, d$	$\emptyset$	$\emptyset$

$$\begin{aligned}
\delta^*(a, 0100110) &= \bigcup_{r\in\delta(a,0)} \delta^*(r, 100110) = \bigcup_{r\in\{a\}} \delta^*(r, 100110) = \delta^*(a, 100110)\\
&= \bigcup_{r\in\delta(a,1)} \delta^*(r, 00110) = \bigcup_{r\in\{a,b\}} \delta^*(r, 00110) = \delta^*(a, 00110) \cup \delta^*(c, 00110)\\
&= \bigcup_{r\in\delta(a,0)} \delta^*(r, 0110) \cup \bigcup_{r\in\delta(c,0)} \delta^*(r, 0110)
\end{aligned}$$

$$\begin{aligned}
&= \bigcup_{r\in\{a\}} \delta^*(r,0110) \cup \bigcup_{r\in\{d\}} \delta^*(r,0110) \\
&= \delta^*(a,0110) \cup \delta^*(d,0110) \\
&= \bigcup_{r\in\delta(a,0)} \delta^*(r,110) \cup \bigcup_{r\in\delta(d,0)} \delta^*(r,110) \\
&= \bigcup_{r\in\{a\}} \delta^*(r,110) \cup \bigcup_{r\in\emptyset} \delta^*(r,110) = \delta^*(a,110) \cup \emptyset \\
&= \bigcup_{r\in\delta(a,1)} \delta^*(r,10) = \bigcup_{r\in\{a,b\}} \delta^*(r,10) = \delta^*(a,10) \cup \delta^*(c,10) \\
&= \bigcup_{r\in\delta(a,1)} \delta^*(r,0) \cup \bigcup_{r\in\delta(c,1)} \delta^*(r,0) \\
&= \bigcup_{r\in\{a,b\}} \delta^*(r,0) \cup \bigcup_{r\in\{d\}} \delta^*(r,0) \\
&= \delta^*(a,0) \cup \delta^*(b,0) \cup \delta^*(d,0) \\
&= \bigcup_{r\in\delta(a,0)} \delta^*(r,\epsilon) \cup \bigcup_{r\in\delta(b,0)} \delta^*(r,\epsilon) \cup \bigcup_{r\in\delta(d,0)} \delta^*(r,\epsilon) \\
&= \bigcup_{r\in\{a\}} \delta^*(r,\epsilon) \cup \bigcup_{r\in\{c,d\}} \delta^*(r,\epsilon) \cup \bigcup_{r\in\emptyset} \delta^*(r,\epsilon) \\
&= \delta^*(a,\epsilon) \cup \delta^*(c,\epsilon) \cup \delta^*(d,\epsilon) \cup \emptyset \\
&= \{a,c,d\}
\end{aligned}$$

es decir, el AFND acepta la palabra, dado que en el conjunto calculado se encuentra un estado final del AFND.

No acepta ϵ, dado que $\delta^*(a,\epsilon) = \{a\}$ y $\{a\} \cap \{d\} = \emptyset$.

5. Hoja 4 (16 de Marzo de 2010)

P1: Convierte el AFND de la hoja anterior en un autómata finito determinista. Incluye en tu solución la tabla de conversión tal como lo vimos en clase, la quíntupla del AFD obtenido finalmente, y su grafo.

R1: La tabla del AFND es:

δ	0	1
$\Longrightarrow a$	$\{a\}$	$\{a,b\}$
b	$\{c,d\}$	$\{c\}$
c	$\{d\}$	$\{d\}$
$\star\, d$	$\emptyset$	$\emptyset$

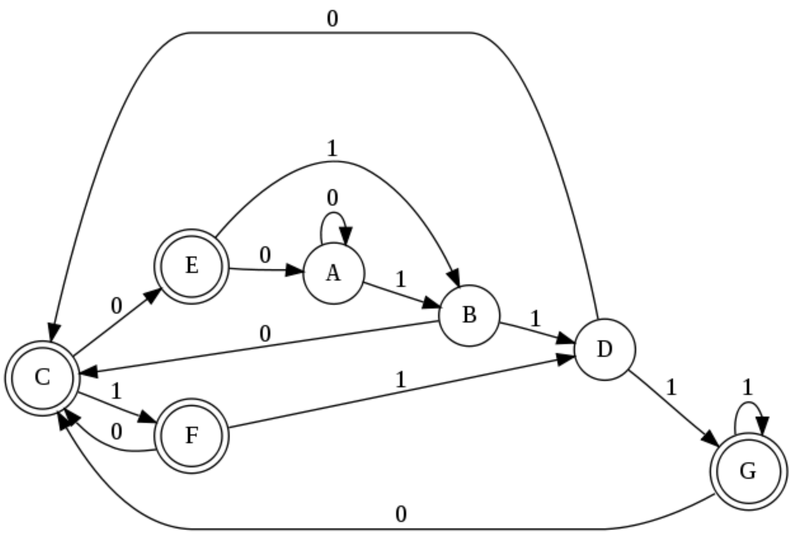
Convertimos a AFD (obviando de las llaves para los conjuntos):

δ	0	1
$\Longrightarrow a$	a	a,b
a,b	a,c,d	a,b,c
$\star\, a,c,d$	a,d	a,b,d
a,b,c	a,c,d	a,b,c,d
$\star\, a,d$	a	a,b
$\star\, a,b,d$	a,c,d	a,b,c
$\star\, a,b,c,d$	a,c,d	a,b,c,d

y con estados renombrados

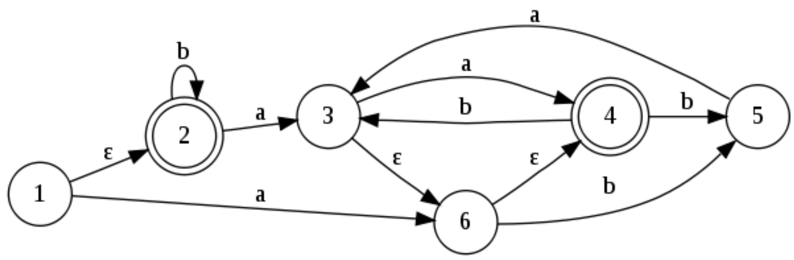
δ	0	1
$\Longrightarrow A$	A	B
B	C	D
$\star\, C$	E	F
D	C	G
$\star\, E$	A	B
$\star\, F$	C	D
$\star\, G$	C	G

La quíntupla es $M_{\mathrm{AFD}} = (\{0,1\}, \{A,B,C,D,E,F,G\}, \delta, A, \{C,E,F,G\})$ con δ según la tabla arriba y el grafo (sin estado inicial marcado, se ha hecho con la herramienta graphviz):



6. Hoja 5 (23 de Marzo de 2010)

P1: Convierte el siguiente AFND-ε en un AFND (incluye todas las tablas, grafos, quíntuplas, y el cálculo de la clausura transitiva necesario). El estado inicial es el estado 1.



R1: Quíntupla del AFND-ε: $M = (\{a,b\},\{1,2,3,4,5,6\},\delta,1,\{2,4\})$, con δ según tabla:

δ	a	b	ε
$\Longrightarrow 1$	$\{6\}$	$\emptyset$	$\{2\}$
$\star\, 2$	$\{3\}$	$\{2\}$	$\emptyset$
3	$\{4\}$	$\emptyset$	$\{6\}$
$\star\, 4$	$\emptyset$	$\{3,5\}$	$\emptyset$
5	$\{3\}$	$\emptyset$	$\emptyset$
6	$\emptyset$	$\{5\}$	$\{4\}$

Cálculo de la clausura-ε:

$$
\begin{aligned}
T_0 &= \{(1,1),(2,2),(3,3),(4,4),(5,5),(6,6)\} \\
T_1 &= \{(1,2),(3,6),(6,4)\} \\
T_2 &= \{(3,4)\} \\
T_3 &= \emptyset
\end{aligned}
$$

$$
T^\star = \bigcup_{i=0}^{\infty} T_i = \{(1,1),(1,2),(2,2),(3,3),(3,4),(3,6),(4,4),(5,5),(6,4),(6,6)\}
$$

Con eso las tablas (concatenadas) de transformación a AFND:

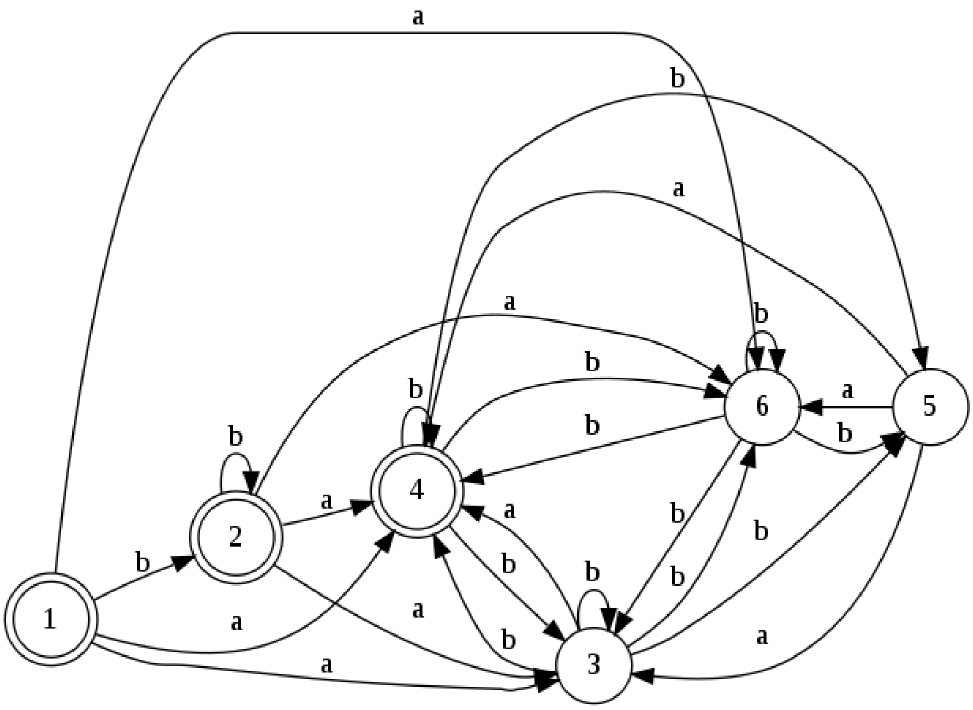
δ	a	b	ε	cl_ε	a	b	a	b
1	$\{6\}$	$\emptyset$	$\{2\}$	$\{1,2\}$	$\{3,6\}$	$\{2\}$	$\{3,4,6\}$	$\{2\}$
2	$\{3\}$	$\{2\}$	$\emptyset$	$\{2\}$	$\{3\}$	$\{2\}$	$\{3,4,6\}$	$\{2\}$
3	$\{4\}$	$\emptyset$	$\{6\}$	$\{3,4,6\}$	$\{4\}$	$\{3,5\}$	$\{4\}$	$\{3,4,5,6\}$
4	$\emptyset$	$\{3,5\}$	$\emptyset$	$\{4\}$	$\emptyset$	$\{3,5\}$	$\emptyset$	$\{3,4,5,6\}$
5	$\{3\}$	$\emptyset$	$\emptyset$	$\{5\}$	$\{3\}$	$\emptyset$	$\{3,4,6\}$	$\emptyset$
6	$\emptyset$	$\{5\}$	$\{4\}$	$\{4,6\}$	$\emptyset$	$\{3,5\}$	$\emptyset$	$\{3,4,5,6\}$

¡Se observa que hay un estado final de M en la clausura-ε del estado inicial, por eso hay que aumentar el conjunto de estados finales!

Resulta la quíntupla del AFND: $M' = (\{a, b\}, \{1, 2, 3, 4, 5, 6\}, \delta, 1, \{1, 2, 4\})$, con δ según tabla:

δ	a	b
$\Longrightarrow \star\, 1$	$\{3, 4, 6\}$	$\{2\}$
$\star\, 2$	$\{3, 4, 6\}$	$\{2\}$
3	$\{4\}$	$\{3, 4, 5, 6\}$
$\star\, 4$	$\emptyset$	$\{3, 4, 5, 6\}$
5	$\{3, 4, 6\}$	$\emptyset$
6	$\emptyset$	$\{3, 4, 5, 6\}$

y el grafo:



7. Hoja 6 (6 de Abril de 2010)

P1: ¿Qué lenguajes definen las siguientes expresiones regulares?

1. $(((0.1)^* + (0.1)^*.0) + ((1.0)^* + (1.0)^*.1))$
2. $(1 + \varepsilon).(0.1)^*.(0 + \varepsilon)$
3. $(0.1 + (((\varepsilon)^* + (\emptyset.0)^*) + 0.1)^*.1)$

R1:

1. La E.R. define todas las palabras donde se alternan 0's y 1's incluyendo la palabra ε.
2. La E.R. define todas las palabras donde se alternan 0's y 1's incluyendo la palabra ε.
3. La E.R. define todas las palabras del conjunto: $L = \{1, 01, 011, 01011, 0101011, 010101011, \ldots\}$ dado que

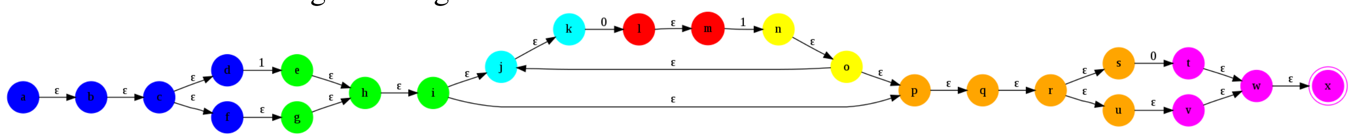
$$\begin{aligned} & (0.1 + (((\varepsilon)^* + (\emptyset.0)^*) + 0.1)^*.1) \\ = & (0.1 + ((\varepsilon + \emptyset) + 0.1)^*.1) \\ = & (0.1 + (0.1)^*.1) \end{aligned}$$

P2: Escribe una expresión regular sobre el alfabeto $\{a, b, c\}$ que define todas las palabras que tengan dos a's consecutivas o dos b's consecutivas.

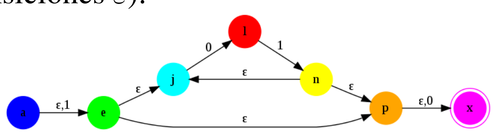
R2: Si interpretamos como *por lo menos* dos a's consecutivas o *por lo menos* dos b's consecutivas, una E.R. es: $(((a+b)+c))^*.(a.a+b.b).(((a+b)+c))^*$ que se puede escribir también como $(a+b+c)^*(aa+bb)(a+b+c)^*$ (obviando de los puntos y de algunas parentesis).

P3: Construye según procedimiento visto en clase un AFND-ε que acepta las mismas palabras que define la expresión regular de **P1** apartado *2*.

R3: El autómata según las reglas vistas en clase es:



Como se ve sin grandes dificultades, se puede unir los estados del mismo color en uno (eliminando así muchas de las transiciones ε):



8. Hoja 7 (13 de Abril de 2010)

P1: Determina una expresión regular α que define el mismo lenguaje aceptado por el autómata de la Hoja 5, digamos M, es decir, se tiene que cumplir $L(\alpha) = L(M)$. Documenta tu construcción suficientemente (tienes dos lados de una hoja).

R1: Primero tenemos que construir un AFD a partir del AFND (usamos para la contrucción de la tabla del AFD como nombres de los estados la concatenación de cifras correspondientes y renombramos después convenientemente):

AFND

δ	a	b
$\Longrightarrow \star\ 1$	$\{3,4,6\}$	$\{2\}$
$\star\ 2$	$\{3,4,6\}$	$\{2\}$
3	$\{4\}$	$\{3,4,5,6\}$
$\star\ 4$	$\emptyset$	$\{3,4,5,6\}$
5	$\{3,4,6\}$	$\emptyset$
6	$\emptyset$	$\{3,4,5,6\}$

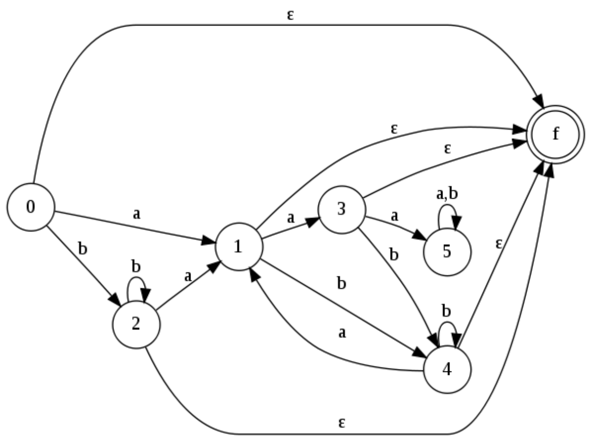
AFD

δ	a	b
$\Longrightarrow \star\ 1$	346	2
$\star\ 346$	4	3456
$\star\ 2$	346	2
$\star\ 4$	$\emptyset$	3456
$\star\ 3456$	346	3456
$\emptyset$	$\emptyset$	$\emptyset$

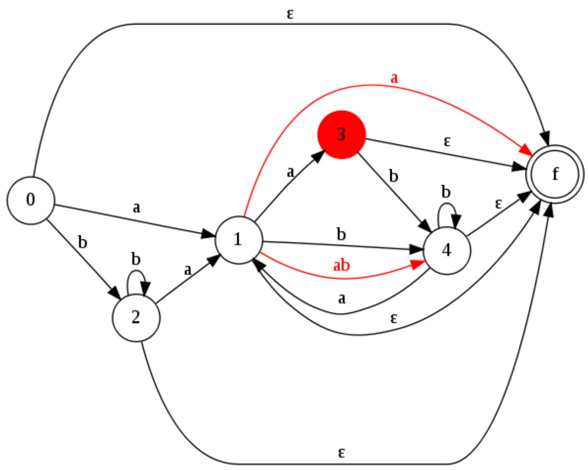
AFD (renombrado)

δ	a	b
$\Longrightarrow \star\ 0$	1	2
$\star\ 1$	3	4
$\star\ 2$	1	2
$\star\ 3$	5	4
$\star\ 4$	1	4
5	5	5

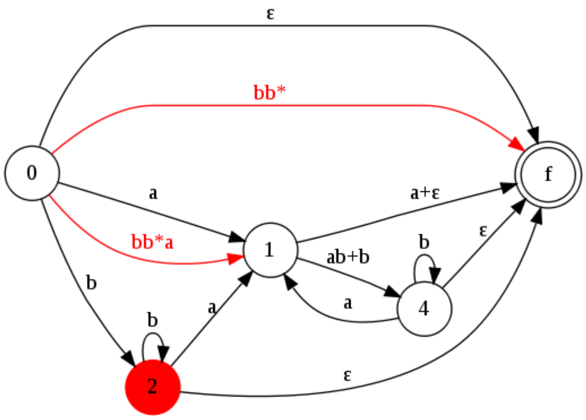
Se obtiene el grafo ya añadido el nuevo estado final f y modificados los estados finales originales (siendo 0 el estado inicial):



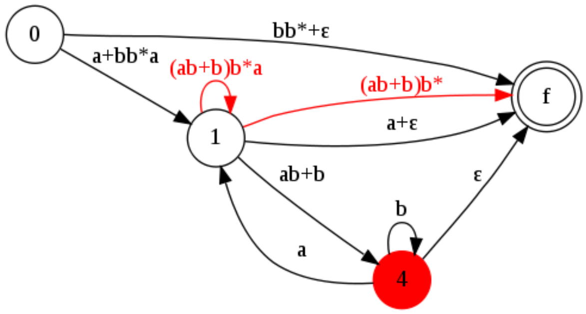
Ya eliminado estado 5 porque no tiene arista saliente a otro estado y añadido las aristas para eliminar estado 3, se obtiene:



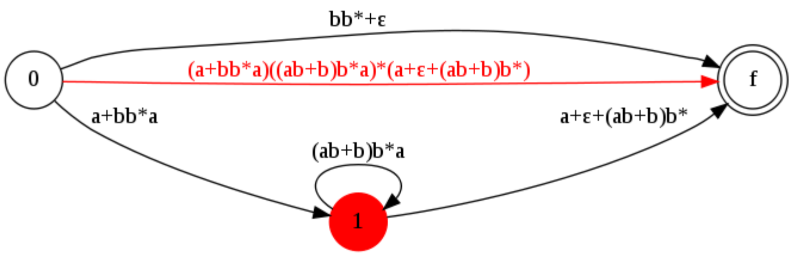
Eliminado estado 3, unidas las aristas paralelas, y añadido las aristas para eliminar estado 2, se obtiene:



Eliminado estado 2, unidas las aristas paralelas, y añadidos las aristas para eliminar estado 4, se obtiene:



Eliminado estado 4, unidas las aristas paralelas, y añadido las aristas para eliminar estado 1, se obtiene:



Eliminado estado 1, unidas las aristas paralelas, se obtiene la expresión regular:

$$bb^* + \varepsilon + (a + bb^*a)((ab + b)b^*a)^*(a + \varepsilon + (ab + b)b^*)$$

9. Hoja 8 (20 de Abril de 2010)

P1: Sea $C = \{a, b, c, d, e, f, g, h\}$ un conjunto (de letras), y sea

$$R = \{(a, d), (b, e), (c, a), (d, b), (e, c), (f, g), (g, h)\} \subset C \times C$$

una relación sobre C.

- Construye las relaciones R^i para todos los $i = 0, \ldots, \infty$.
- Construye la relacion R^*.
- Argumenta si R^* es reflexiva, simétrica, y/o transitiva.
- ¿Cuál pareja (o parejas) 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:

$$
\begin{aligned}
R^0 &= \{(a, a), (b, b), (c, c), (d, d), (e, e), (f, f), (g, g), (h, h)\} && (1)\\
R^1 &= \{(a, d), (b, e), (c, a), (d, b), (e, c), (f, g), (g, h)\} && (2)\\
R^2 &= \{(a, b), (b, c), (c, d), (d, e), (e, a), (f, h)\} && (3)\\
R^3 &= \{(a, e), (b, a), (c, b), (d, c), (e, d)\} && (4)\\
R^4 &= \{(a, c), (b, d), (c, e), (d, a), (e, b)\} && (5)\\
R^5 &= \{(a, a), (b, b), (c, c), (d, d), (e, e)\} && (6)\\
R^6 &= \{(a, d), (b, e), (c, a), (d, b), (e, c)\} && (7)\\
R^7 &= \{(a, b), (b, c), (c, d), (d, e), (e, a)\} && (8)\\
& \text{para } n > 7 && \\
R^8 &= R^3 && (9)\\
R^9 &= R^4 && (10)\\
R^{10} &= R^5 && (11)\\
R^{11} &= R^6 && (12)\\
R^{12} &= R^7 && (13)\\
& \text{es decir} && \\
R^n &= R^{(n-3) \bmod 5 + 3} && (14)
\end{aligned}
$$

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

$$\begin{aligned} R^* \;=\; & \{(a,a),(a,b),(a,c),(a,d),(a,e),\\ & (b,a),(b,b),(b,c),(b,d),(b,e),\\ & (c,a),(c,b),(c,c),(c,d),(c,e),\\ & (d,a),(d,b),(d,c),(d,d),(d,e),\\ & (e,a),(e,b),(e,c),(e,d),(e,e),\\ & (f,f),(f,g),(f,h),(g,g),(g,h),(h,h)\} \end{aligned} \tag{15}$$

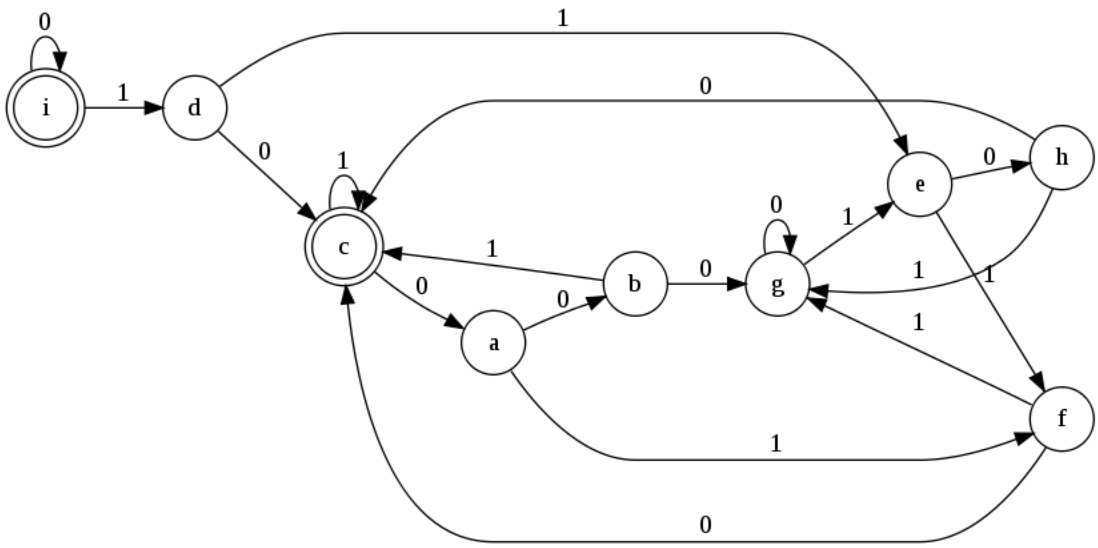
- **reflexiva:** sí, dado que $R^0 \subset R^*$
  simétrica: no, dado que $(f,h) \in R^*$ pero $(h,f) \notin R^*$
  transitiva: sí, por construcción.

- Se añade (h,f) y se verifica fácilmente que R^* es simétrica:

$$\begin{aligned} R^* \;=\; & \{(a,a),(a,b),(a,c),(a,d),(a,e),\\ & (b,a),(b,b),(b,c),(b,d),(b,e),\\ & (c,a),(c,b),(c,c),(c,d),(c,e),\\ & (d,a),(d,b),(d,c),(d,d),(d,e),\\ & (e,a),(e,b),(e,c),(e,d),(e,e),\\ & (f,f),(f,g),(f,h),\\ & (g,f),(g,g),(g,h),\\ & (h,f),(h,g),(h,h)\} \end{aligned} \tag{16}$$

10. Hoja 9 (27 de Abril de 2010)

P1: Minimiza el siguiente autómata finito determinista. El estado inicial es el estado a.

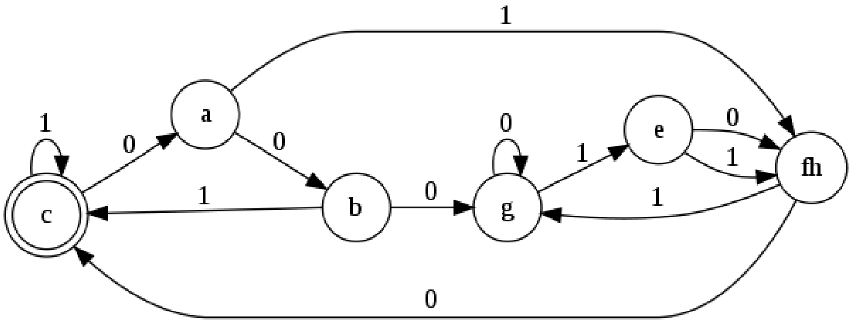


R1: Los estados i y d no son alcanzables desde el estado inicial a, se eliminan. Se observa que el autómata restanto está completo y conexo. La tabla de minimización se desarrolla como sigue con los siguientes comentarios *i)* los índices indican en que paso se ha marcado la casilla, *ii)* en la segunda tabla se resume el análisis de las parejas, las parejas en negritas son aquellas que ya han sido marcados, *iii)* la columna de *marca* indica la marca puesta para la pareja correspondiente, *iv)* la columna de *lista* contiene las entradas pendientes cuando en algún caso no se marca nada:

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

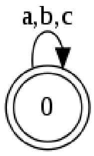
pareja	0	1	marca	lista
(a,b)	(b,g)	$(\mathbf{f},\mathbf{c})$	X_1	
(a,e)	(b,h)	(f,f)	X_7	
(a,f)	$(\mathbf{b},\mathbf{c})$	(f,g)	X_2	
(a,g)	(b,g)	(f,e)	X_6	
(a,h)	$(\mathbf{b},\mathbf{c})$	(f,g)	X_3	
(b,e)	(g,h)	$(\mathbf{c},\mathbf{f})$	X_4	
(b,f)	$(\mathbf{g},\mathbf{c})$	$(\mathbf{c},\mathbf{g})$	X_5	
(b,g)	(g,g)	$(\mathbf{c},\mathbf{e})$	X_6	(a,g)
(b,h)	$(\mathbf{g},\mathbf{c})$	$(\mathbf{c},\mathbf{g})$	X_7	(a,e)
(e,f)	$(\mathbf{h},\mathbf{c})$	(f,g)	X_8	(a,g)
(e,g)	(h,g)	$(\mathbf{f},\mathbf{e})$	X_9	
(e,h)	$(\mathbf{h},\mathbf{c})$	(f,g)	X_{10}	
(f,g)	$(\mathbf{c},\mathbf{g})$	$(\mathbf{g},\mathbf{e})$	X_{11}	
(f,h)	(c,c)	(g,g)		
(g,h)	$(\mathbf{g},\mathbf{c})$	$(\mathbf{e},\mathbf{g})$	X_{12}	

Entonces la pareja (f, h) marca dos estados equivalentes y el AFD mínimo es (con estado inicial a):

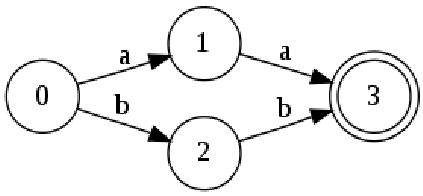


P2: Determina el $Indice(R_L)$ del lenguaje definido por una expresión regular sobre el alfabeto $\{a, b, c\}$ que define todas las palabras que tengan por lo menos dos a's consecutivas o por lo menos dos b's consecutivas (mira hoja 6 si quieres).

R2: Para hallar dicho $Indice(R_L)$ construimos un AFD mínimo equivalente a la expresión regular $(a+b+c)^*(aa+bb)(a+b+c)^*$ (cual define el lenguaje). Se observa que se pueden construir fácilmente automátas para las tres partes de la E.R. siendo 0 el estado inicial (la primera es igual a la tercera):

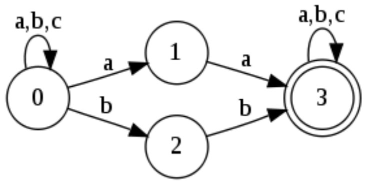


$(a+b+c)^*$



$aa+bb$

Concatenando los tres autómatas correspondientes y uniendo ya los estados unidos por transiciones ε tal como visto en la hoja 6, se obtiene un AFND equivalente a la E.R.:



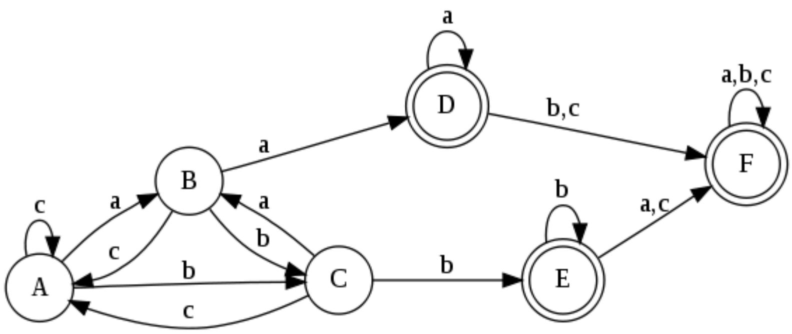
Un AFD equivalente obtenemos con la construcción (obviando de las llaves):

δ	a	b	c
$\Longrightarrow 0$	$0,1$	$0,2$	0
$0,1$	$0,1,3$	$0,2$	0
$0,2$	$0,1$	$0,2,3$	0
$\star\, 0,1,3$	$0,1,3$	$0,3$	$0,3$
$\star\, 0,2,3$	$0,3$	$0,2,3$	$0,3$
$\star\, 0,3$	$0,3$	$0,3$	$0,3$

y con estados renombrados

δ	a	b	c
$\Longrightarrow A$	B	C	A
B	D	C	A
C	B	E	A
$\star\, D$	D	F	F
$\star\, E$	F	E	F
$\star\, F$	F	F	F

cuyo grafo (según construcción conexo y todo acesible) es



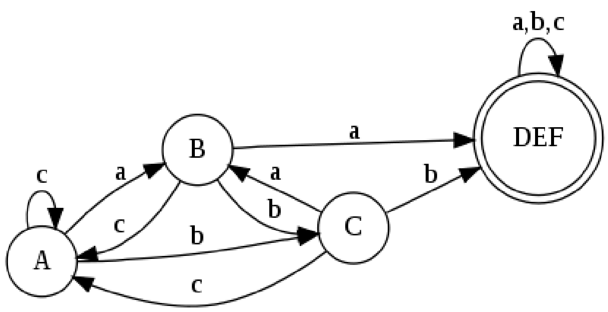
la tabla de minimización se obtiene

	A	B	C	$\star D$	$\star E$	$\star F$
A	-	X_1	X_2	X_0	X_0	X_0
B	-	-	X_3	X_0	X_0	X_0
C	-	-	-	X_0	X_0	X_0
$\star D$	-	-	-	-		
$\star E$	-	-	-	-	-	
$\star F$	-	-	-	-	-	-

con la siguiente análisis de parejas

pareja	a	b	c	marca	lista
(A, B)	$(\mathbf{B}, \mathbf{D})$	(C, C)	(A, A)	X_1	
(A, C)	(B, B)	$(\mathbf{C}, \mathbf{E})$	(A, A)	X_2	
(B, C)	$(\mathbf{D}, \mathbf{B})$	$(\mathbf{C}, \mathbf{E})$	(A, A)	X_3	
(D, E)	(D, F)	(F, E)	(F, F)		
(D, F)	(D, F)	(F, F)	(F, F)		(D, E)
(E, F)	(F, F)	(E, F)	(F, F)		(D, E)

Se observa que los estados D, E, y F son equivalentes entre si, y el AFD mínimo es:

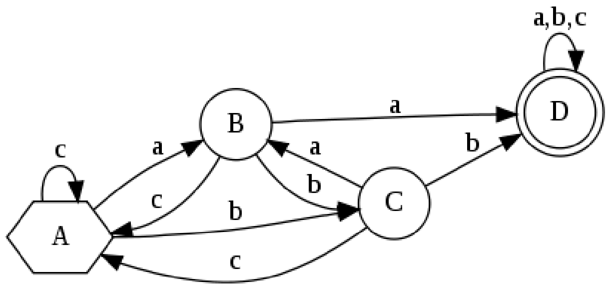


y concluimos que el $Indice(R_L)$ es 4.

11. Hoja 10 (4 de Mayo de 2010)

P1: Construye una gramática lineal por la izquierda que genera el mismo lenguaje que define la expresión regular $(a+b+c)^*(aa+bb)(a+b+c)^*$. (Ayuda: puedes aprovechar de resultados de la hoja 9).

R1: Usamos el AFD de la hoja 9 (con el estado DEF renombrado y usando un hexágono para marcar el estado inicial):



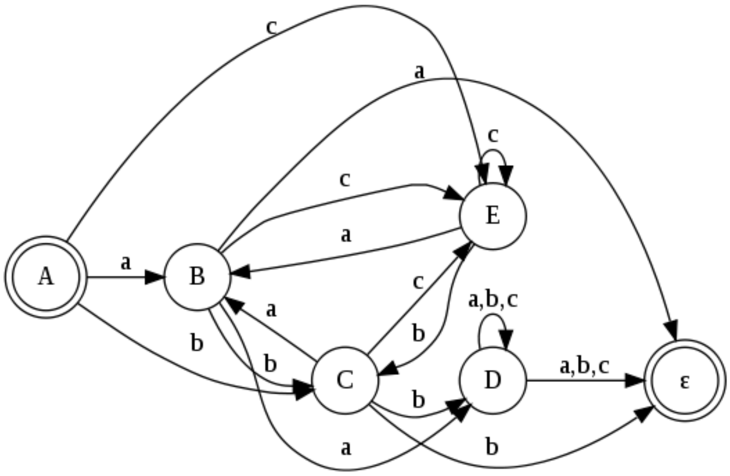
De ahí leemos directamente una gramática lineal por la derecha:

$$
\begin{aligned}
G &= (\Sigma_N, \Sigma_T, P, \$) \\
&= (\{A, B, C, D\}, \{a, b, c\}, P, A) \\
P &= \{A \longrightarrow aB \mid bC \mid cA, B \longrightarrow aD \mid a \mid bC \mid cA, C \longrightarrow aB \mid bD \mid b \mid cA, \\
&\quad D \longrightarrow aD \mid a \mid bD \mid b \mid cD \mid c\}
\end{aligned}
$$

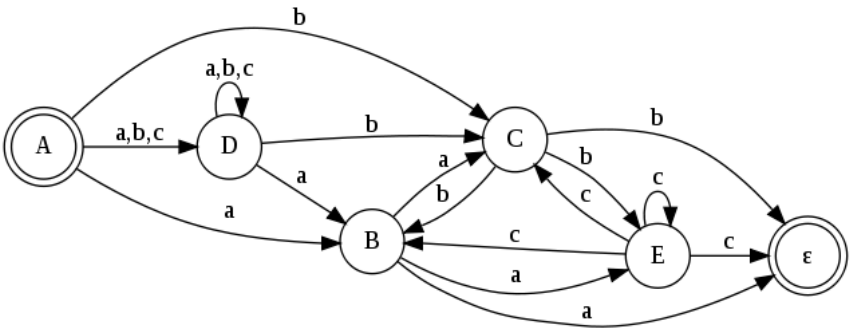
Observamos que el símbolo inicial A aparece a la derecha de algunas producciones, por eso añadimos un nuevo símbolo E para sustituirlo:

$$
\begin{aligned}
G &= (\Sigma_N, \Sigma_T, P, \$) \\
&= (\{A, B, C, D, E\}, \{a, b, c\}, P, A) \\
P &= \{A \longrightarrow aB \mid bC \mid cE, B \longrightarrow aD \mid a \mid bC \mid cE, C \longrightarrow aB \mid bD \mid b \mid cE, \\
&\quad D \longrightarrow aD \mid a \mid bD \mid b \mid cD \mid c, E \longrightarrow aB \mid bC \mid cE, \}
\end{aligned}
$$

Construimos a partir de la gramática el grafo correspondiente:



Invertimos el grafo:



Y leemos la gramática lineal por la izquierda resultante:

$$
\begin{aligned}
G &= (\{A, B, C, D, E\}, \{a, b, c\}, P, A) \\
P &= \{A \longrightarrow Ba \mid Cb \mid Da \mid Db \mid Dc, B \longrightarrow Ca \mid Ea \mid a, C \longrightarrow Bb \mid Eb \mid b \\
& \quad\; D \longrightarrow Ba \mid Cb \mid Da \mid Db \mid Dc, E \longrightarrow Bc \mid Cc \mid Ec \mid c\}
\end{aligned}
$$

12. Hoja 11 (11 de Mayo de 2010)

P1: Construye una gramática libre de contexto G que genera todas las palabras sobre el alfabeto $\{a, b\}$ que tienen el mismo número de a's y b's, es decir,

$$L(G) = \{\varepsilon, ab, ba, aabb, abab, baab, abba, baba, bbaa, aaabbb, aababb, \ldots\}$$

R1: Usamos un argumento recursivo para encontrar la gramática. Si $w = \varepsilon$, entonces tiene cero a's y cero b's (entonces usamos la producción $\$ \longrightarrow \varepsilon$). Sea w una palabra con tantas a's como b's y $|w| > 0$. Entonces podemos subdividir w en cuatro partes: o bien en $w = axby$ o bien en $w = bxay$ donde tanto x como y son palabras (más cortas que w) que tienen tantas a's como b's. Nota que x y/o y pueden ser ε.

Eso nos lleva a la siguiente gramática:

$$\begin{aligned} G &= (\{\$\}, \{a, b\}, P, \$) \\ P &= \{\$ \longrightarrow a\$b\$ \mid b\$a\$ \mid \varepsilon\} \end{aligned}$$

P2: Construye una gramática libre de contexto G que genera todas las palabras sobre el alfabeto $\{a, b\}$ que tienen un número diferente de a's y b's (ayuda: amplía la gramática del primer punto).

R2: Ya sabemos como generar palabras con tantas a's como b's. Usamos de nuevo un argumento recursivo.

Si el número debe ser diferente, entonces la palabra tendrá o bien más a's o bien más b's. Miramos el primer caso: sea w una palabra que tiene más a's que b's. Entonces podemos subdivir w en tres partes: $w = xay$ donde x es una palabra con un número igual de a's y b's y y es una palabra o bien con un número igual de a's y b's o bien con más a's que b's. En el segundo caso es lo mismo con b: sea w una palabra que tiene más b's que a's. Entonces podemos subdivir w en tres partes: $w = xby$ donde x es una palabra con un número igual de a's y b's y y es una palabra o bien con un número igual de a's y b's o bien con más b's que a's. Nota que tanto x como y son más cortas que w en ambos casos.

Entonces usamos tres variables: una variable I para producir palabras con un número igual de a's y b's (tal como en **R1** con \$) y una variable A para el primer caso y otra variable B para el segundo caso, es decir, la gramática es:

$$\begin{aligned} G &= (\{\$, A, B, I\}, \{a, b\}, P, \$) \\ P &= \{\$ \longrightarrow A \mid B, A \longrightarrow IaI \mid IaA, B \longrightarrow IbI \mid IbB, I \longrightarrow aIbI \mid bIaI \mid \varepsilon\} \end{aligned}$$

13. Hoja 12 (18 de Mayo de 2010)

P1: Construye un autómata finito de pila no-determinista (AFPND) que acepta las mismas palabras que genera la gramática de la hoja 11, apartado 2.

R1: Construimos un AFPND que acepta en estado final. La idea principal es: usamos la pila para *contar* la diferencia entre el número de a's y b's. Para distinguir los casos de tener más a's que b's o, al revés, más b's que a's usamos los estados.

Entonces necesitamos 4 estados que llamamos A, B, 0, y f:

- Si estamos en el estado A (contando las a's demás):
  - apilamos cada a en la pila, y
  - para cada b quitamos una a de la pila, si hay tal a, y
  - si encontramos una b sin a en la pila, apilamos la b y vamos al estado B (ya que tenemos una b demás).
- Si estamos en el estado B (contando las b's demás):
  - apilamos cada b en la pila, y
  - para cada a quitamos una b de la pila, si hay tal b, y
  - si encontramos una a sin b en la pila, apilamos la a y vamos al estado A (ya que tenemos una b demás).
- Usamos el estado 0 como estado inicial para realizar una transición a A o a B dependiendo si el primer símbolo es una a o una b.
- Si estamos en A y hay una a en la pila hacemos una transición ε al estado final.
- Si estamos en B y hay una b en la pila hacemos una transición ε al estado final.

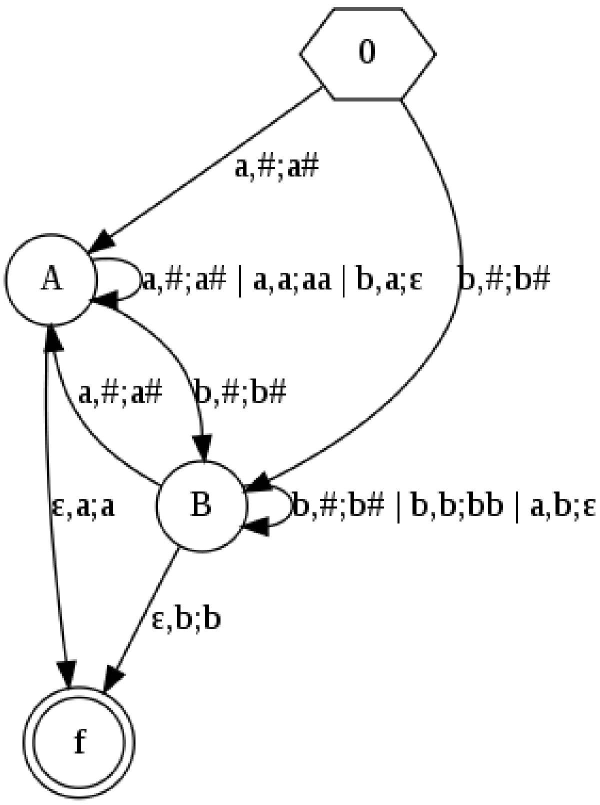
Con estos argumentos el AFPND será:

$$M = (\{a, b\}, \{\#, a, b\}, \{0, A, B, f\}, \delta, 0, \#, \{f\})$$

$$\begin{aligned}
\delta(0, a, \#) &= (A, a\#) \\
\delta(0, b, \#) &= (B, b\#)
\end{aligned}$$

$$\begin{aligned}
\delta(A, a, \#) &= (A, a\#) \\
\delta(A, a, a) &= (A, aa) \\
\delta(A, b, \#) &= (B, b\#)
\end{aligned}$$

$$\begin{aligned}
\delta(A,b,a) &= (A,\varepsilon)\\
\delta(A,\varepsilon,a) &= (f,a)\\
\\
\delta(B,b,\#) &= (B,b\#)\\
\delta(B,b,b) &= (B,bb)\\
\delta(B,a,\#) &= (A,a\#)\\
\delta(B,a,b) &= (B,\varepsilon)\\
\delta(B,\varepsilon,b) &= (f,b)
\end{aligned}$$



Hemos usado la barra vertical para separar las diferentes transciones posibles entre parejas de estados. El hexágono marca el estado inicial.

Observa: al final, la pila contiene justamente la diferencia de símbolos, es decir, o bien las a's o bien las b's que "sobran".

Este AFPND es una posibilidad, hay muchísimas más posibilidades incluso transformaciones *algorítmicas* de gramática libre de contexto a autómata de pila.

14. Hoja 13 (1 de Junio de 2010)

P1: Transforma la gramática de la hoja 11, apartado 2, en forma normal de Chomsky.

Esta hoja no hay que entregar.