Teoría de Autómatas y Lenguajes Formales

Ejercicios de Lenguajes Regulares

Autores:

Araceli Sanchis de Miguel
Agapito Ledezma Espino
Jose A. Iglesias Martínez
Beatriz García Jiménez
Juan Manuel Alonso Weber

* Algunos ejercicios están basados en enunciados de los siguientes libros:

- Enrique Alfonseca Cubero, Manuel Alfonseca Cubero, Roberto Moriyón Salomón. *Teoría de autómatas y lenguajes formales*. McGraw-Hill (2007).
- Manuel Alfonseca, Justo Sancho, Miguel Martínez Orga. *Teoría de lenguajes, gramáticas y autómatas*. Publicaciones R.A.E.C. (1997).
- Pedro Isasi, Paloma Martínez y Daniel Borrajo. *Lenguajes, Gramáticas y Autómatas. Un enfoque práctico.* Addison-Wesley (1997).

1. **Obtener el AFD mínimo equivalente a cada una de las siguientes gramáticas, describiendo las transformaciones intermedias: G → AFND → AFD → AFD mínimo.**

a) G_A = ({a,b,c}, {S,A,B}, S, P_A) P_A = { S ::= aA | bB | c A ::= aB | b | cA B ::= a | bA | c }	**b)** G_B = ({a,b,c}, {S,B,C,E}, S, P_B) P_B = { S ::= a | aS | aB | cC C ::= c B ::= bE | b E ::= bB | b }
c) G_C = ({a,b,c}, {S,A,B,C,D}, S, P_C) P_C = { S ::= aA A ::= aA | bB | a B ::= bB | bC | b C ::= bC | cD | bB D ::= bC | bB |cC }	**d)** G_D = ({c,f,d}, {A,B,C,D,E,F}, A, P_D) P_D = { A ::= cB | cE | f | fC B ::= cB | fD | dE C ::= cA D ::= cD | fD E ::= cE | fF | dF F ::= cF | fF }

2. **Sea el alfabeto de terminales {a,b}. Desarrollar un autómata que reconozca cadenas de longitud "3" del lenguaje universal). Obtener la G3 correspondiente a dicho Autómata.**

3. **Una puerta blindada dispone de una única cerradura. Para abrirla es necesario hacer girar en ella tres llaves diferentes (denominadas a, b y c), en un orden predeterminado, que se describe a continuación:**
   - **Llave a, seguida de llave b, seguida de llave c, o bien**
   - **Llave b, seguida de llave a, seguida de llave c.**

   Si no se respeta este orden, la puerta se bloquea, y es imposible su apertura; por ejemplo, si se hace girar la llave a, se retira la misma, se introduce de nuevo y se hace girar.

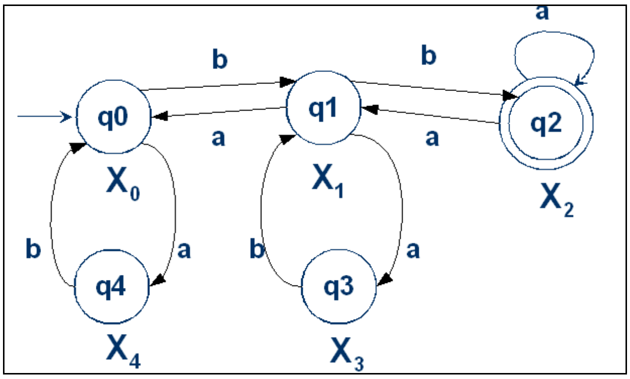
   Una vez abierta la puerta, la introducción de las llaves en su cerradura, en cualquier orden, no afecta al mecanismo de cierre (la puerta permanece abierta).

   Considérese que las denominaciones de las llaves son símbolos de un alfabeto, sobre el que define el lenguaje L cuyas palabras son las secuencias permitidas de apertura de la puerta. Por ejemplo, abcbc es una palabra del referido lenguaje.

   Se pide:
   a) **Diseño del autómata AF finito que acepta L.**
   b) **Gramática (limpia y bien formada) que genera las palabras de L.**

4. **Dada la ER (b•a*)* que representa a un lenguaje regular, construir un AF que acepte ese lenguaje regular.**

5. **Hallar el lenguaje reconocido por el siguiente autómata utilizando ecuaciones características.**



6. **Dada la siguiente gramática regular lineal derecha, G= ({0,1},{S,A,B,C},S,P): donde P={ S::= 1A | 1B, A ::= 0A | 0C | 1C | 1 , B ::= 1A | 1C | 1 , C ::= 1}. Se pide hallar, formalmente la Expresión Regular del lenguaje asociado a dicha gramática.**

7. **Simplifique la siguiente expresión regular:** $\alpha = a + a(b+aa)(b^*aa)^*b^* + a(aa+b)^*$ **utilizando las propiedades de equivalencia de las expresiones regulares.**

8. **Calcular la derivada $D_{ab}(\alpha)$ siendo α = a*ab, utilizando las definiciones de derivadas de expresiones regulares.**

9. **Obtenga la gramática para la expresión regular** $a(aa + b)^*$.

10. **Dada la siguiente expresión regular** $a^*c^*(a+b)\ (cb)^*$**, construir formalmente la gramática regular equivalente.**

SOLUCIONES

1. Obtener el AFD mínimo equivalente a cada una de las siguientes gramáticas, describiendo las transformaciones intermedias: G → AFND → AFD.

e) G_A = ({a,b,c}, {S,A,B}, S, P_A) P_A = { S ::= aA | bB | c A ::= aB | b | cA B ::= a | bA | c }	**f)** G_B = ({a,b,c}, {S,B,C,E}, S, P_B) P_B = { S ::= a | aS | aB | cC C ::= c B ::= bE | b E ::= bB | b }
g) G_C = ({a,b,c}, {S,A,B,C,D}, S, P_C) P_C = { S ::= aA A ::= aA | bB | a B ::= bB | bC | b C ::= bC | cD | bB D ::= bC | bB |cC }	**h)** G_D = ({c,f,d}, {A,B,C,D,E,F}, A, P_D) P_D = { A ::= cB | cE | f | fC B ::= cB | fD | dE C ::= cA D ::= cD | fD E ::= cE | fF | dF F ::= cF | fF }

Solución:

IMPORTANTE: El paso de AFD a AFD mínimo es igual que el Ejercicio 5 de Tema 3. Lo novedoso en este tema es el paso de G a AF y de AFND a AFD. De esta forma se pueden enlazar unos temas con otros, para dar lugar a un problema más completo.

a) AFND + sumidero = AFD = AFDmin

AFND	AFD	AFD mínimo
AFND=({a,b,c},{Q0,Q1,Q2,Q3},f,Q0,Q3) f(Q0, a) = Q1 f(Q0, b) = Q2 f(Q0, c) = Q3 f(Q1, a) = Q2 f(Q1, b) = Q3 f(Q1, c) = Q1 f(Q2, a) = Q3 f(Q2, b) = Q1 f(Q2, c) = Q3	AFD=({a,b,c},{Q0,Q1,Q2,Q3,Q4},f,Q0,Q3) f(Q0, a) = Q1 f(Q0, b) = Q2 f(Q0, c) = Q3 f(Q1, a) = Q2 f(Q1, b) = Q3 f(Q1, c) = Q1 f(Q2, a) = Q3 f(Q2, b) = Q1 f(Q2, c) = Q3 f(Q3, a) = Q4 f(Q3, b) = Q4 f(Q3, c) = Q4 f(Q4, a) = Q4 f(Q4, b) = Q4 f(Q4, c) = Q4	Igual que AFD.

b) AFND <> AFD <> AFDmin

- 2 Estados finales, equivalentes entre sí, formados por 2 estados cada uno.

AFND	AFD	AFD mínimo
AFND=({a,b,c},{Q0,Q1,Q2,Q3,Q7},f,Q0,Q3) f(Q0, a) = Q0,Q2,Q3 f(Q0, c) = Q1 f(Q1, c) = Q3 f(Q2, b) = Q2,Q3 f(Q7, b) = Q2,Q3	AFD=({a,b,c},{Q0,Q1,Q3,Q4,Q5,Q6,Q8},f,Q0,{Q3,Q4,Q6,Q8}) f(Q0, a) = Q4 f(Q0, b) = Q5 f(Q0, c) = Q1 f(Q1, a) = Q5 f(Q1, b) = Q5 f(Q1, c) = Q3 f(Q3, a) = Q5 f(Q3, b) = Q5 f(Q3, c) = Q5 f(Q4, a) = Q4 f(Q4, b) = Q8 f(Q4, c) = Q1 f(Q5, a) = Q5 f(Q5, b) = Q5 f(Q5, c) = Q5 f(Q6, a) = Q5 f(Q6, b) = Q8 f(Q6, c) = Q5 f(Q8, a) = Q5 f(Q8, b) = Q6 f(Q8, c) = Q5	AFDmin=({a,b,c},{Q0,Q1,Q3,Q4,Q5,Q9},f,Q0,{Q3,Q4,Q9}) f(Q0, a) = Q4 f(Q0, b) = Q5 f(Q0, c) = Q1 f(Q1, a) = Q5 f(Q1, b) = Q5 f(Q1, c) = Q3 f(Q3, a) = Q5 f(Q3, b) = Q5 f(Q3, c) = Q5 f(Q4, a) = Q4 f(Q4, b) = Q9 f(Q4, c) = Q1 f(Q5, a) = Q5 f(Q5, b) = Q5 f(Q5, c) = Q5 f(Q9, a) = Q5 f(Q9, b) = Q9 f(Q9, c) = Q5
Correspondencia con G S ≡ Q0 B ≡ Q2 C ≡ Q1 E ≡ Q7 Final: Q3	Correspondencia con AFND {Q0,Q2,Q3} ≡ Q4 {Q2,Q3} ≡ Q6 {Q7,Q3} ≡ Q8 El resto se mantienen igual.	Correspondencia con AFD {Q6,Q8} ≡ Q9 El resto se mantienen igual.

c) AFND <> AFD <> AFDmin

- Se crean estados combinados que no son estado final (porque no contienen al final del AFND) (Q6).
- Desaparece el estado final del AFND.
- Sólo aparecen 2 nuevos estados finales.
- Combinación de 2 y otros de 3 estados.

AFND	**AFD**	**AFD mínimo**
AFND=({a,b,c},{Q0,Q1,Q2,Q3,Q4,Q5},f,Q0,Q5) f(Q0, a) = Q1 f(Q1, a) = Q1,Q5 f(Q1, b) = Q2 f(Q2, b) = Q2,Q3,Q5 f(Q3, b) = Q2,Q3 f(Q3, c) = Q4 f(Q4, b) = Q2,Q3 f(Q4, c) = Q3	AFD=({a,b,c},{Q0,Q1,Q2,Q3,Q4,Q6,Q7,Q8,Q9},f,Q0,{Q7,Q8}) f(Q0, a) = Q1 f(Q0, b) = Q6 f(Q0, c) = Q6 f(Q1, a) = Q7 f(Q1, b) = Q2 f(Q1, c) = Q6 f(Q7, a) = Q7 f(Q7, b) = Q2 f(Q7, c) = Q6 f(Q2, a) = Q6 f(Q2, b) = Q8 f(Q2, c) = Q6 f(Q8, a) = Q6 f(Q8, b) = Q8 f(Q8, c) = Q4 f(Q4, a) = Q6 f(Q4, b) = Q9 f(Q4, c) = Q3 f(Q9, a) = Q6 f(Q9, b) = Q8 f(Q9, c) = Q4 f(Q3, a) = Q6 f(Q3, b) = Q9 f(Q3, c) = Q4 f(Q6, a) = Q6 f(Q6, b) = Q6 f(Q6, c) = Q6	AFDmin=({a,b,c},{Q0,Q1,Q2,Q6,Q7,Q8,Q9,Q10},f,Q0,{Q7,Q8}) f(Q0, a) = Q1 f(Q0, b) = Q6 f(Q0, c) = Q6 f(Q1, a) = Q7 f(Q1, b) = Q2 f(Q1, c) = Q6 f(Q7, a) = Q7 f(Q7, b) = Q2 f(Q7, c) = Q6 f(Q2, a) = Q6 f(Q2, b) = Q8 f(Q2, c) = Q6 f(Q8, a) = Q6 f(Q8, b) = Q8 f(Q8, c) = Q10 f(Q9, a) = Q6 f(Q9, b) = Q8 f(Q9, c) = Q10 f(Q6, a) = Q6 f(Q6, b) = Q6 f(Q6, c) = Q6 f(Q10, a) = Q6 f(Q10, b) = Q9 f(Q10, c) = Q10
Correspondencia con G S ≡ Q0 A ≡ Q1 B ≡ Q2 C ≡ Q3 D ≡ Q4 Final : Q5	Correspondencia con AFND {Q1,Q5} ≡ Q7 {Q2,Q3,Q5} ≡ Q8 {Q2,Q3} ≡ Q9 El resto se mantienen igual.	Correspondencia con AFD {Q3,Q4} ≡ Q10 El resto se mantienen igual.

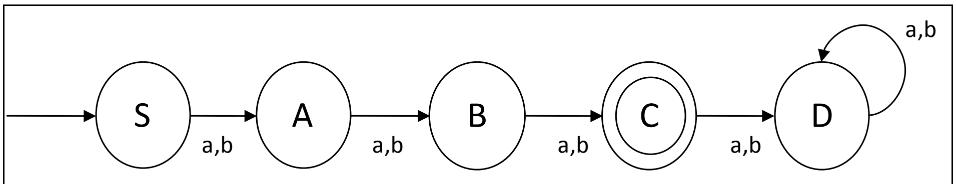
d) AFND <> AFD <> AFDmin

- Se eliminan muchos estados de AFND a AFD.
- Se reduce mucho de AFD a AFDmin (con solo 3 iteraciones).
- El AFDmin sólo tiene 3 estados: inicial, final y sumidero.

AFND	**AFD**	**AFD mínimo**
`AFND=({c,f,d},{Q0,Q1,Q2,Q3,Q4,Q5,Q6},f,Q0,Q6)` f(Q0, c) = Q1,Q4 f(Q0, f) = Q2,Q6 f(Q1, c) = Q1 f(Q1, f) = Q3 f(Q1, d) = Q4 f(Q2, c) = Q0 f(Q3, c) = Q3 f(Q3, f) = Q3 f(Q4, c) = Q4 f(Q4, f) = Q5 f(Q4, d) = Q5 f(Q5, c) = Q5 f(Q5, f) = Q5	`AFD=({c,f,d},{Q0,Q5,Q7,Q8,Q9,Q10,Q11},f,Q0,Q8)` f(Q0, c) = Q7 f(Q0, f) = Q8 f(Q0, d) = Q9 f(Q7, c) = Q7 f(Q7, f) = Q10 f(Q7, d) = Q11 f(Q8, c) = Q0 f(Q8, f) = Q9 f(Q8, d) = Q9 f(Q10, c) = Q10 f(Q10, f) = Q10 f(Q10, d) = Q9 f(Q11, c) = Q11 f(Q11, f) = Q5 f(Q11, d) = Q5 f(Q5, c) = Q5 f(Q5, f) = Q5 f(Q5, d) = Q9 f(Q9, c) = Q9 f(Q9, f) = Q9 f(Q9, d) = Q9	`AFDmin=({c,f,d},{C1,C2,C3},f,C3,C2)` f(C3, c) = C1 f(C3, f) = C2 f(C3, d) = C1 f(C1, c) = C1 f(C1, f) = C1 f(C1, d) = C1 f(C2, c) = C3 f(C2, f) = C1 f(C2, d) = C1
Correspondencia con G A ≡ Q0 B ≡ Q1 C ≡ Q2 D ≡ Q3 E ≡ Q4 F ≡ Q5 Final: Q6	Correspondencia con AFND Q0 ≡ Q0 {Q1,Q4} ≡ Q7 {Q2,Q6} ≡ Q8 {Q3,Q5} ≡ Q10 {Q4,Q5} ≡ Q11 Q5 ≡ Q5 Q9 ≡ Q9	Correspondencia con AFD Q0 ≡ C3 Q8 ≡ C2 {Q5,Q7,Q9,Q10,Q11} ≡ C1

2. Sea el alfabeto de terminales {a,b}. Obtener la gramática de tipo 3 que genere cadenas de longitud tres sobre dicho alfabeto a través del Autómata Finito que las reconozca.

Solución:



S::=aA | bA	**Gramática Bien Formada**
A::=aB | bB	S::=aA | bA
B::=aC | bC	A::=aB | bB
B::=a | b	B::=a | b
C::=aD | bD	
D::=aD | bD	

3. Una puerta blindada dispone de una única cerradura. Para abrirla es necesario hacer girar en ella tres llaves diferentes (denominadas a, b y c), en un orden predeterminado, que se describe a continuación:

- **Llave a, seguida de llave b, seguida de llave c, o bien**
- **Llave b, seguida de llave a, seguida de llave c.**

Si no se respeta este orden, la puerta se bloquea, y es imposible su apertura; por ejemplo, si se hace girar la llave a, se retira la misma, se introduce de nuevo y se hace girar.

Una vez abierta la puerta, la introducción de las llaves en su cerradura, en cualquier orden, no afecta al mecanismo de cierre (la puerta permanece abierta).

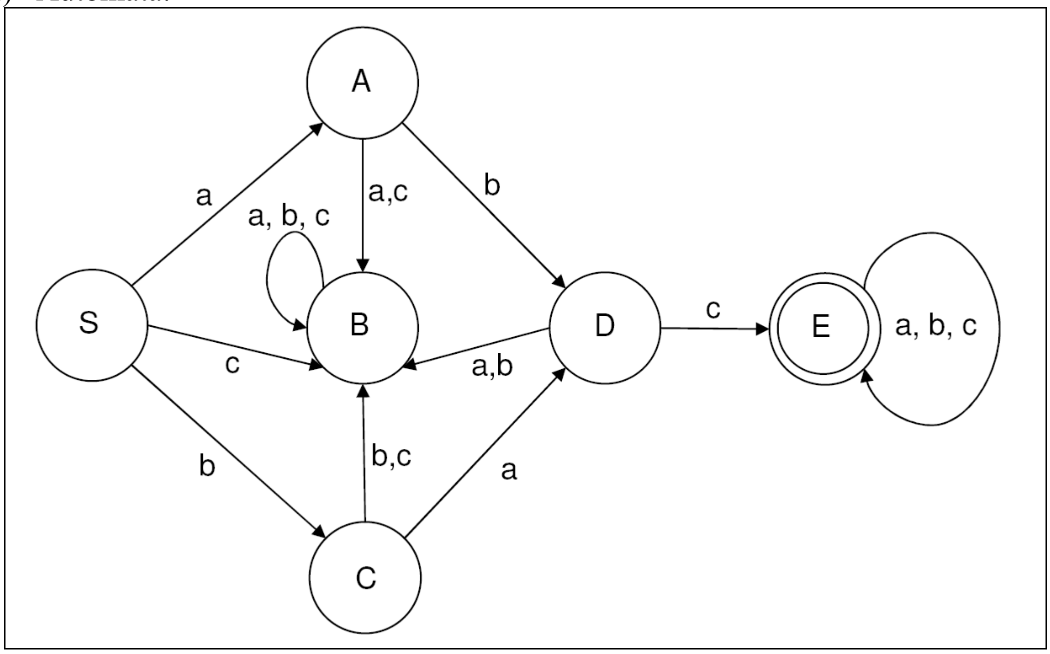
Considérese que las denominaciones de las llaves son símbolos de un alfabeto, sobre el que define el lenguaje L cuyas palabras son las secuencias permitidas de apertura de la puerta. Por ejemplo, abcbc es una palabra del referido lenguaje.

Se pide:

a) Diseño del autómata AF finito que acepta L.
b) Gramática (limpia y bien formada) que genera las palabras de L.

Solución:

a) Autómata:



b) Gramática bien formada:

G = ({a, b, c}, {S, A, B, C, D, E}, S, P)

P = { S ::= aA | bC
A ::= bD
C ::= aD
D ::= cE | c
E ::= aE | bE | cE | a | b | c
}

4. Dada la ER (b•a*)* que representa a un lenguaje regular, construir un AF que acepte ese lenguaje regular.

Solución:

a. <u>Con derivadas (ER → G → AF):</u>

$R = (ba^*)^*$

$$
\begin{aligned}
D_a(R) &= D_a((ba^*)^*) = D_a(ba^*)(ba^*)^* \\
&= (\cancel{D_a(b)}a^* + \cancel{\delta(b)}D_a(a^*))\ (ba^*)^* \\
&= (\phi a^* + \phi D_a(a^*))\ (ba^*)^* && = \phi
\end{aligned}
$$

$$
\begin{aligned}
D_b(R) &= D_b((ba^*)^*) = D_b(ba^*)(ba^*)^* \\
&= (D_b(b)a^* + \cancel{\delta(b)}D_b(a^*))\ (ba^*)^* \\
&= (\lambda a^* + \phi D_b(a^*))\ (ba^*)^* = \mathbf{a^*(ba^*)^*} = \mathbf{R1}
\end{aligned}
$$

$$
\begin{aligned}
D_a(R1) &= D_a(a^*(ba^*)^*) \\
&= D_a(a^*)(ba^*)^* + \delta(a^*)D_a((ba^*)^*) \\
&= D_a(a)a^*(ba^*)^* + \lambda \cancel{D_a(R)} \\
&= \lambda a^*(ba^*)^* + \lambda\phi = a^*(ba^*)^* = R1
\end{aligned}
$$

$$
\begin{aligned}
D_b(R1) &= D_b(a^*(ba^*)^*) \\
&= \cancel{D_b(a^*)}(ba^*)^* + \delta(a^*)D_b((ba^*)^*) \\
&= \phi(ba^*)^* + \lambda D_b(R) = \phi + R1 = R1
\end{aligned}
$$

No se obtienen más EERR. **R = (ba*)*** **R1 = a*(ba*)***

$D_a(R)$	=	ϕ	$\delta(D_a(R))$	$= \delta(\phi)$		=	ϕ
$D_b(R)$	=	R1	$\delta(D_b(R))$	$= \delta(R1) =$	$\delta(a^*(ba^*)^*))$	=	λ
$D_a(R1)$	=	R1	$\delta(D_a(R1))$	$= \delta(R1) =$	$\delta(a^*(ba^*)^*))$	=	λ
$D_b(R1)$	=	R1	$\delta(D_b(R1))$	$= \delta(R1) =$	$\delta(a^*(ba^*)^*))$	=	λ

Las producciones serían:

P = {

$R \rightarrow bR1$ $R \rightarrow b$

$R1 \rightarrow aR1$ $R1 \rightarrow a$

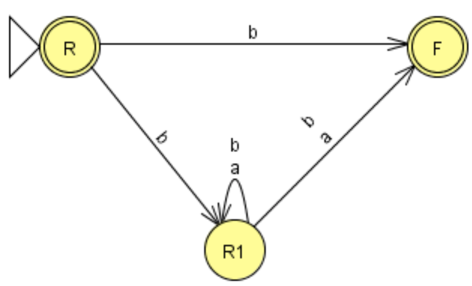
$R1 \rightarrow bR1$ $R1 \rightarrow b$

$R \rightarrow \lambda$ puesto que $\delta(R) = \lambda$

}

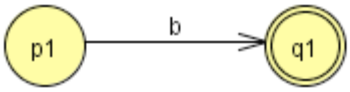
G = ({a,b}, {R, R1}, R, P)

AF = ({a,b}, {R, R1, F}, f, R, {F})

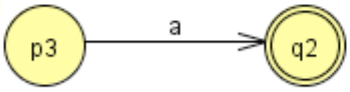


b. <u>Con algoritmo de síntesis recursivo (composición de grafos):</u>

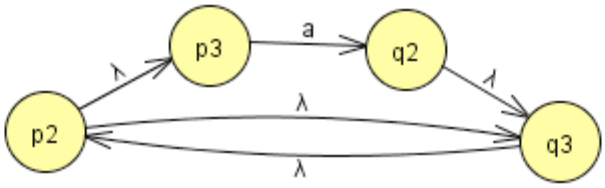
b:



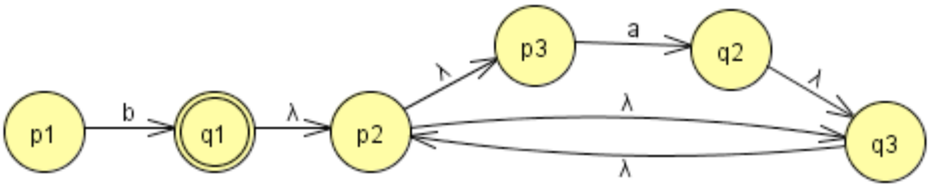
a:



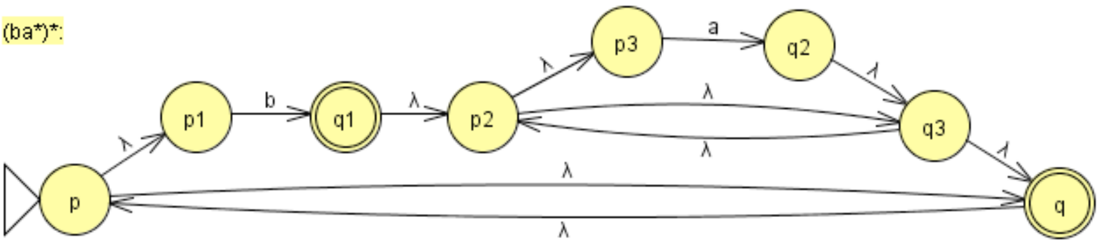
a*:



ba*:

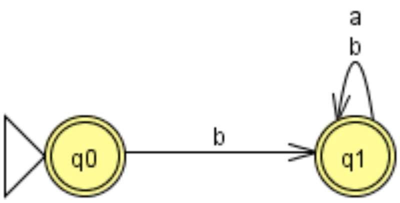


(ba*)*:

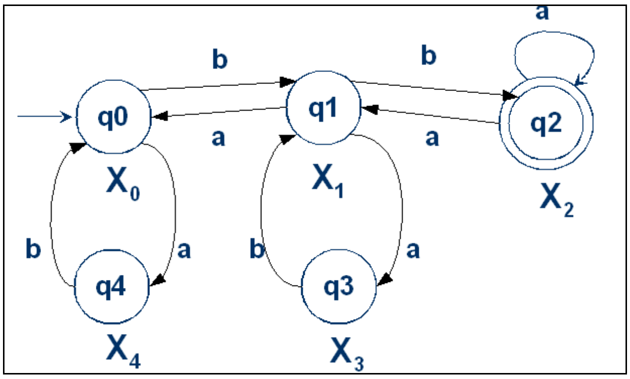


c. <u>Diseñándolo directamente:</u>

AF=({a,b},{q0,q1},f,q0,{q1}), f:



5. Hallar el lenguaje reconocido por el siguiente autómata utilizando ecuaciones características.



Solución:

X0 = aX4 + bX1
X1 = aX0 + bX2 + b + aX3
X2 = aX1+ aX2 + a
X3 = bX1
X4 = bX0

a. Empezar por las ecuaciones alejadas de q0

X2 = aX1 + aX2 + a → X2 = a* • (aX1 + a)

b. Continuar acercándonos a q0 (X0)

X1 = a X0+ b X2 + b + aX3 → X1=a X0+ b • (a* • (aX1 + a)) + b + a • bX1 = (ba*a + ab) • X1 + aX0 +b a*a+b

X1 = (b a* a + a b)* • (aX0 + b a* a + b)

X0 = a X4+ b X1 → X0 = a b X0 + b • ((b a* a + a b)* • (aX0 + b a* a + b))

X0 = a b X0 + b • (b a* a + a b)* • aX0 + b • (b a* a + a b)* • b a*a + b • (b a* a + a b)*• b

X0 = X0• (ab + ab • (ba*a + ab)*) + b • (ba*a + ab)* ba*a + b (ba*a + ab)* b

X0 = (ab + b) • (ba*a + ab)* • a)* • (b • (ba*a + ab)* • (ba*a + b))

X0 = (ab)* b (ab)* b (ab)* a* es el lenguaje reconocido

6. Dada la siguiente gramática regular lineal derecha, G= ({0,1},{S,A,B,C},S,P): donde P={ S::= 1A | 1B, A ::= 0A | 0C | 1C | 1 , B ::= 1A | 1C | 1 , C ::= 1}. Se pide hallar, formalmente la Expresión Regular del lenguaje asociado a dicha gramática.

Solución:

De la gramática G = {{0, 1}, {S, A, B, C}, S, P)

con P = { $S \rightarrow 1A \mid 1B$

$A \rightarrow 0A \mid 0C \mid 1C \mid 1$

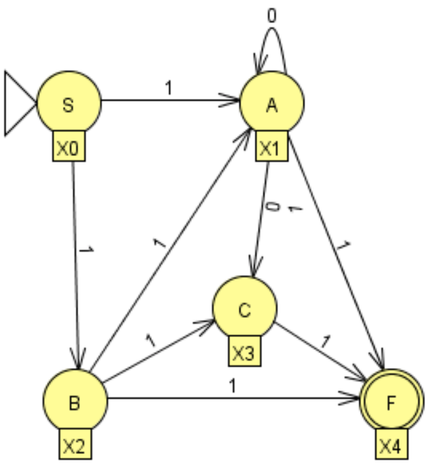
$B \rightarrow 1A \mid 1C \mid 1$

$C \rightarrow 1$

}

Obtenemos un AFND.

	0	1
→S		A,B
A	A,C	C,F
B		A,C,F
C		F
*F		



Para obtener las ecuaciones características renombramos los estados a X_i

$X_0 = 1 \cdot X_1 + 1 \cdot X_2$

$X_1 = 0 \cdot X_1 + 0 \cdot X_3 + 1 \cdot X_3 + 1 \cdot X_4$

$X_2 = 1 \cdot X_1 + 1 \cdot X_3 + 1 \cdot X_4$

$X_3 = 1X_4 \quad \Rightarrow \quad X_3 = 1$

$X_4 = \lambda$

$X_1 \quad = 0 \cdot X_1 + 0 \cdot 1 + 1 \cdot 1 + 11 \qquad = 0 \cdot X_1 + (0 \cdot 1 + 1 \cdot 1 + 1)$

$A = 0,\ B = (0 \cdot 1 + 1 \cdot 1 + 1) \ \Rightarrow \ X_1 \ = 0^*(01 + 11 + 1)$

$X_2 \quad = 10^*(0 \cdot 1 + 1 \cdot 1 + 1) + 1 \cdot 1 + 1 \qquad = 10^*(0 \cdot 1 + 1 \cdot 1 + 1) + 1 \cdot 1 + 1 =$

$X_0 \quad = 1 \cdot 0^*(0 \cdot 1 + 1 \cdot 1 + 1) + 1 \cdot 1 \cdot 0^*(0 \cdot 1 + 1 \cdot 1 + 1) \ + 1 \cdot 1 + 1 \quad =$

$= (\lambda \ + 1)(10^*(01 + 11 + 1)) + 11 + 1$

ER= (λ+1)(10*(01+11+1))+11+1

7. Simplifique la siguiente expresión regular: $\alpha = a + a(b+aa)(b^*aa)^*b^* + a(aa+b)^*$ utilizando las propiedades de equivalencia de las expresiones regulares.

Solución:

a. Aplicando la propiedad: $(\alpha + \beta)^* = (\alpha^* \beta^*)^* \alpha^*$

$\alpha = a + a(b + aa)\ (b^*aa)^*b^* + a(aa + b)^*$
$\alpha = a + a(b + aa)(b + aa)^* + a(aa + b)^*$

b. Aplicando la propiedad 4, sacando factor común a *a*.

$\alpha = a + a(b + aa)(b + aa)^* + a(aa + b)^*$
$\alpha = a(\lambda + (b + aa)(b + aa)^*) + a(aa + b)^*$

c. Aplicando la propiedad: $\alpha^* = \lambda + \alpha\alpha^*$

$\alpha = a(\lambda + (b + aa)(b + aa)^* + a(aa + b)^*$
$\alpha = a(b + aa)^* + a(aa + b)^*$

d. Aplicando la propiedad : $\alpha + \beta = \beta + \alpha$

$\alpha = a\ (b + aa)^* + a(aa + b)^*$
$\alpha = a(aa + b)^* + a(aa + b)^*$

e. Aplicando, si α y β son ER => $L(\alpha + \beta) = L(\alpha)\ U\ L(\beta)$

$\alpha = a(aa + b)^* + a(aa + b)^* = a(aa + b)^*$

8. Calcular la derivada $D_{ab}(\alpha)$ siendo α = a*ab, utilizando las definiciones de derivadas de expresiones regulares.

Solución:

$D_{ab}(\alpha) = D_b(D_a(\alpha))$,

derivamos primero respecto de a:

$D_a(\alpha) = D_a(a^*)\cdot ab + \delta(a^*){:}D_a(ab)$

$D_a(\alpha) = D_a(a)a^*ab + \delta(D_a(a)b + \delta(a)D_a(b))$

$D_a(\alpha) = a^*ab + b$

$D_a(\alpha) = (a^*a + \lambda)b$

aplicando la propiedad $\alpha^* = \lambda + \alpha\alpha^*$:

$D_a(\alpha) = a^*b$ (sin esta simplificación el resultado final es el mismo)

derivamos ahora respecto de b: $D_b(D_a(\alpha))=D_b(a^*b)$:

$D_b(a^*b) = D_b(a^*)\cdot b + \delta(a^*)\cdot D_b(b) = \varnothing\cdot b + \lambda\cdot\lambda = \varnothing + \lambda = \lambda$

$\mathbf{D_{ab}(a^*ab) = \lambda}$

9. Obtenga la gramática para la expresión regular *a(aa + b).**

Solución:

Resultado de las derivaciones sucesivas:
$D_a(R_0) = (aa+b)^* = R1$

$D_b(R_0) = \phi$

$D_a(R_1) = a(aa+b)^* = R_0$

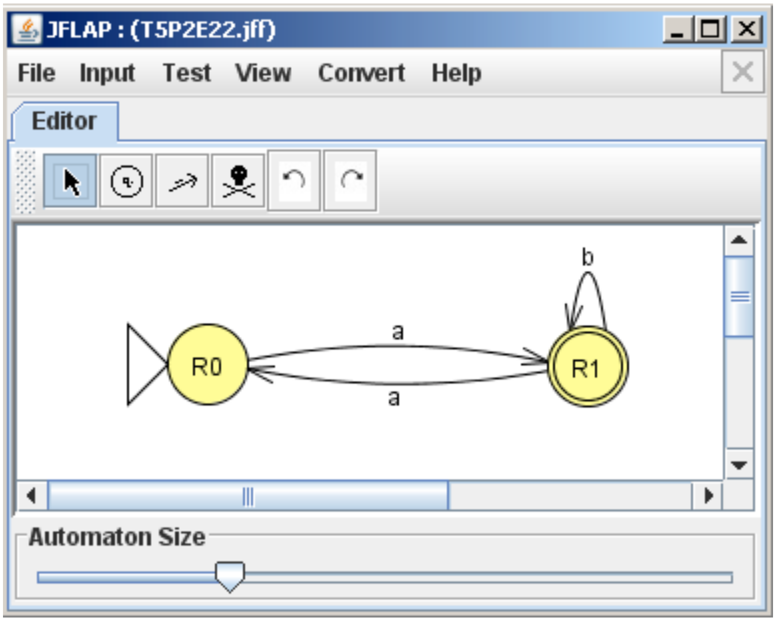
$D_b(R_1) = (aa+b)^* = R_1$

Resumiendo:
$R_0 = a(aa+b)^*$
$R_1 = (aa+b)^*$

$D_a(R_0) = R_1$	$R_0 ::= aR_1$	$\delta(D_a(R_0)) = \lambda$	$R_0 ::= a$
$D_b(R_0) = \phi$			
$D_a(R_1) = R0$	$R_1 ::= aR_0$	$\delta(D_a(R_1)) = \phi$	
$D_b(R_1) = R_1$	$R_1 ::= bR_1$	$\delta(D_b(R_1)) = \lambda$	$R_1 ::= b$

$G = (\{a, b\}, \{R_0, R_1\}, R_0 , P)$
P = {
$R_0 ::= aR_1 \mid a$
$R_1 ::= aR_0$
$R_1 ::= bR_1 \mid R_1 ::= b$
}

El AFD equivalente es:



10. Dada la siguiente expresión regular *a*c*(a+b) (cb), construir formalmente la gramática regular equivalente.**

Solución:

Resultado de las derivaciones sucesivas:

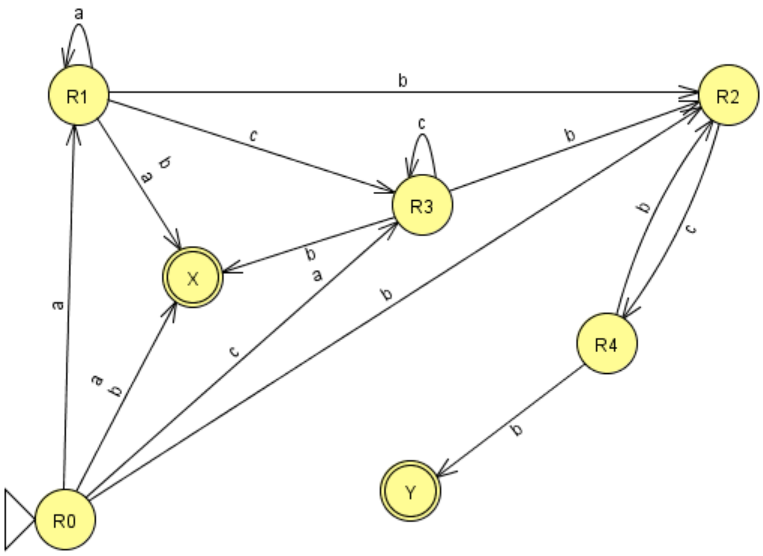
R_0 = a*c*(a+b)(cb)*
R_1 = a*c*(a+b)(cb)* + (cb)*
R_2 = (cb)*
R_3 = c*(a+b)(cb)*
R_4 = b(cb)*

Tabla de derivadas:

$D_a(R_0) = R_1$	R_0 -> aR_1	$\delta(D_a(R_0)) = \delta(R_1) = \lambda$	$R_0 \rightarrow a$
$D_b(R_0) = R_2$	R_0 -> bR_2	$\delta(D_b(R_0)) = \delta(R_2) = \lambda$	$R_0 \rightarrow b$
$D_c(R_0) = R_3$	R_0 -> cR_3	$\delta(D_c(R_0)) = \delta(R_3) = \phi$	-
$D_a(R_1) = R_1$	R_1 -> aR_1	$\delta(D_a(R_1)) = \delta(R_1) = \lambda$	$R_1 \rightarrow a$
$D_b(R_1) = R_2$	R_1 -> bR_2	$\delta(D_b(R_1)) = \delta(R_2) = \lambda$	$R_1 \rightarrow b$
$D_c(R_1) = R_3$	R_1 -> cR_3	$\delta(D_c(R_1)) = \delta(R_3) = \phi$	-
$D_a(R_2) = \phi$	-		
$D_b(R_2) = \phi$	-		
$D_c(R_2) = R_4$	R_2 -> cR_4	$\delta(D_c(R_2)) = \delta(R_4) = \phi$	-
$D_a(R_3) = R_2$	R_3 -> aR_2	$\delta(D_a(R_3)) = \delta(R_2) = \lambda$	$R_3 \rightarrow a$
$D_b(R_3) = R_2$	R_3 -> bR_2	$\delta(D_b(R_3)) = \delta(R_2) = \lambda$	$R_3 \rightarrow b$
$D_c(R_3) = R_3$	R_3 -> cR_3	$\delta(D_c(R_3)) = \delta(R_3) = \phi$	-
$D_a(R_4) = \phi$	-		
$D_b(R_4) = R_2$	R_4 -> bR_2	$\delta(D_b(R_4)) = \delta(R_2) = \lambda$	$R_4 \rightarrow b$
$D_c(R_4) = \phi$	-		

Las producciones de la gramática y el autómata finito equivalente son:

R_0 -> aR_1 | a | bR_2 | b | cR_3
R_1 -> aR_1 | a | bR_2 | b | cR_3
R_2 -> cR_4
R_3 -> aR_2 | a | bR_2 | b | cR_3
R_4 -> bR_2 | b



Desarrollo con detalle:

$D_a(R_0) = D_a(a^*c^*(a+b)(cb)^*) = D_a(a^*)c^*(a+b)(cb)^* + \delta(a^*)D_a(c^*(a+b)(cb)^*) =$

$= D_a(a)a^*c^*(a+b)(cb)^*) + D_a(c^*)(a+b)(cb)^* + \delta(c^*)D_a((a+b)(cb)^*) =$

$= \lambda a^*c^*(a+b)(cb)^*) + D_a(c)c^*(a+b)(cb)^* + \lambda D_a(a+b)(cb)^*) =$

$= R_0 + \phi c^*(a+b)(cb)^* + D_a((a+b)(cb)^*) + \delta(a+b)D_a((cb)^*) =$

$= R_0 + D_a(a+b)(cb)^* + \delta(a+b)D_a((cb)^*) + \phi D_a((cb)^*) =$

$= R_0 + D_a(a)(cb)^* + D_a(b)(cb)^* + \phi D_a((cb)^*) =$

$= R_0 + \lambda(cb)^* + \phi(cb)^* = R_0 + (cb)^* = \mathbf{R_1}$

$D_b(R_0) = D_b(a^*c^*(a+b)(cb)^*) = D_b(a^*)c^*(a+b)(cb)^* + \delta(a^*)D_b(c^*(a+b)(cb)^*) =$

$= D_b(a)a^*c^*(a+b)(cb)^*) + D_b(c^*)(a+b)(cb)^* + \delta(c^*)D_b((a+b)(cb)^*) =$

$= \phi a^*c^*(a+b)(cb)^*) + \phi c^*(a+b)(cb)^* + \lambda D_b((a+b)(cb)^*) =$

$= D_b(a+b)(cb)^* + \delta(a+b)D_b((cb)^*) =$

$= D_b(a)(cb)^* + D_b(b)(cb)^* + \phi D_b((cb)^*) = \phi + \lambda(cb)^* = (cb)^* = \mathbf{R_2}$

$D_c(R_0) = D_c(a^*c^*(a+b)(cb)^*) = D_c(a^*)c^*(a+b)(cb)^*) + \delta(a^*)D_c(c^*(a+b)(cb)^*)) =$

$= D_c(a)a^*c^*(a+b)(cb)^*) + D_c(c^*)(a+b)(cb)^* + \delta(c^*)D_c((a+b)(cb)^*) =$

$= \phi a^*c^*(a+b)(cb)^*) + D_c(c)c^*(a+b)(cb)^* + \lambda D_c((a+b)(cb)^*)) =$

$= c^*(a+b)(cb)^* + D_c((a+b))(cb)^* + \delta(a+b)D_c((cb)^*) =$

$= c^*(a+b)(cb)^* + D_c(a)(cb)^* + D_c(b)(cb)^* + \phi D_c((cb)^*) =$

$= c^*(a+b)(cb)^* + \phi(cb)^* + \phi(cb)^* = c^*(a+b)(cb)^* = \mathbf{R_3}$

Resumiendo:

$R_0 = a^*c^*(a+b)(cb)^*$
$R_1 = a^*c^*(a+b)(cb)^* + (cb)^* = R_0 + R_2$ | Aprovechamos la notación más compacta
$R_2 = (cb)^*$
$R_3 = c^*(a+b)(cb)^*$

$D_a(R_0) = R_1$ $D_b(R_0) = R_2$ $D_c(R_0) = R_3$

$D_a(R_1) = D_a(R_0 + R_2) = D_a(R_0) + D_a(R_2) = R_1 + D_a(R_2)$

$D_a(R_1) = D_b(R_0 + R_2) = D_b(R_0) + D_b(R_2) = R_2 + D_b(R_2)$

$D_c(R_1) = D_c(R_0 + R_2) = D_c(R_0) + D_c(R_2) = R_3 + D_c(R_2)$ | Esperamos a derivar R_2

$D_a(R_2) = D_a((cb)^*)$ [1] $= D_a(cb)(cb)^* = D_a(c)b(cb)^* + \delta(c)D_a(b)(cb)^*$

$= \phi b(cb)^* + \phi D_a(b)(cb)^* = \phi$

$D_b(R_2) = D_b((cb)^*) = D_b(cb)(cb)^* = D_b(c)b(cb)^* + \delta(c)D_b(b)(cb)^*$

$= \phi b(cb)^* + \phi D_b(b)(cb)^* = \phi$

$D_c(R_2) = D_c((cb)^*) = D_c(cb)(cb)^* = D_c(c)b(cb)^* + \delta(c)D_c(b)(cb)^*$

$= \lambda b(cb)^* + \phi D_c(b)(cb)^* = b(cb)^* = \mathbf{R_4}$

$D_a(R_1) = D_a(R_0 + R_2) = D_a(R_0) + D_a(R_2) = R_1 + D_a(R_2) = R_1 + \phi = \mathbf{R_1}$

$D_a(R_1) = D_b(R_0 + R_2) = D_b(R_0) + D_b(R_2) = R_2 + D_b(R_2) = R_2 + \phi = \mathbf{R_2}$

$D_c(R_1) = D_c(R_0 + R_2) = D_c(R_0) + D_c(R_2) = R_3 + D_c(R_2) = R_3 + R_4 =$

$= c^*(a+b)(cb)^* + b(cb)^* = c^*a(cb)^* + c^*b(cb)^* + b(cb)^*$ [2] $=$

$= c^*a(cb)^* + (c^* + \lambda)b(cb)^*$ [3] $=$

$= c^*a(cb)^* + c^*b(cb)^* = c^*(a+b)(cb)^* = \mathbf{R_3}$

Resumiendo:

$R_0 = a^*c^*(a+b)(cb)^*$
$R_1 = a^*c^*(a+b)(cb)^* + (cb)^* = R_0 + R_2$
$R_2 = (cb)^*$
$R_3 = c^*(a+b)(cb)^*$
$R_4 = b(cb)^*$

$D_a(R_0) = R_1$	$D_b(R_0) = R_2$	$D_c(R_0) = R_3$
$D_a(R_1) = R_1$	$D_b(R_1) = R_2$	$D_c(R_1) = R_3$
$D_a(R_2) = \phi$	$D_b(R_2) = \phi$	$D_c(R_2) = R_4$

[1] Podemos anticipar el resultado ϕ dado que la cadena a derivar no contiene el símbolo de derivación.
[2] $b(cb)^* \in c^*b(cb)^*$ por lo que podemos anular el tercer término, lo dejamos para el siguiente paso.
[3] $\lambda \in c^*$, por lo que $c^* + \lambda = c^*$, por si queda más claro que en la nota 2.

Proseguimos con las derivadas:

$D_a(R_3) = D_a(c^*(a+b)(cb)^*) = D_a(c^*)(a+b)(cb)^* + \delta(c^*)D_a((a+b)(cb)^*) =$

$= D_a(c)c^*(a+b)(cb)^* + \lambda D_a((a+b))(cb)^* + \delta(a+b)D_a((cb)^*) =$

$= \phi c^*(a+b)(cb)^* + D_a(a)(cb)^* + D_a(b)(cb)^* + \phi D_a((cb)^*) =$

$= \lambda(cb)^* + \phi(cb)^* = (cb)^* = \mathbf{R_2}$

$D_b(R_3) = D_b(c^*(a+b)(cb)^*) = D_b(c^*)(a+b)(cb)^* + \delta(c^*)D_b((a+b)(cb)^*) =$

$= D_b(c)c^*(a+b)(cb)^* + \lambda D_b((a+b))(cb)^* + \delta(a+b)D_b((cb)^*) =$

$= \phi c^*(a+b)(cb)^* + D_b(a)(cb)^* + D_b(b)(cb)^* + \phi D_c((cb)^*) =$

$= \phi(cb)^* + \lambda(cb)^* = (cb)^* = \mathbf{R_2}$

$D_c(R_3) = D_c(c^*(a+b)(cb)^*) = = D_c(c^*)(a+b)(cb)^* + \delta(c^*)D_c((a+b)(cb)^*) =$

$= D_c(c)c^*(a+b)(cb)^* + \lambda D_c((a+b))(cb)^* + \delta(a+b)D_c((cb)^*) =$

$= \lambda c^*(a+b)(cb)^* + D_c(a)(cb)^* + D_c(b)(cb)^* + \phi D_c((cb)^*) =$

$= c^*(a+b)(cb)^* + \phi(cb)^* + \phi(cb)^* = c^*(a+b)(cb)^* = \mathbf{R_3}$

$D_a(R_4) = D_a(b(cb)^*)$ [4] $= D_a(b)(cb)^* + \delta(b)D_a((cb)^*) = \phi(cb)^* + \phi D_a((cb)^*) = \boldsymbol{\phi}$

$D_b(R_4) = D_b(b(cb)^*) = D_b(b)(cb)^* + \delta(b)D_b((cb)^*) = \lambda(cb)^* + \phi D_b((cb)^*)$

$= (cb)^* = \mathbf{R_2}$

$D_c(R_4) = D_c(b(cb)^*) = D_c(b)(cb)^* + \delta(b)D_c((cb)^*) = \phi(cb)^* + \phi D_c((cb)^*) = \boldsymbol{\phi}$

Resumen:

$R_0 = a^*c^*(a+b)(cb)^*$
$R_1 = a^*c^*(a+b)(cb)^* + (cb)^*$
$R_2 = (cb)^*$
$R_3 = c^*(a+b)(cb)^*$
$R_4 = b(cb)^*$

$D_a(R_0) = R_1$	$D_b(R_0) = R_2$	$D_c(R_0) = R_3$
$D_a(R_1) = R_1$	$D_b(R_1) = R_2$	$D_c(R_1) = R_3$
$D_a(R_2) = \phi$	$D_b(R_2) = \phi$	$D_c(R_2) = R_4$
$D_a(R_3) = R_2$	$D_b(R_3) = R_2$	$D_c(R_3) = R_3$
$D_a(R_4) = \phi$	$D_b(R_4) = R_2$	$D_c(R_4) = \phi$

[4] La cadena a derivar no contiene el símbolo de derivación, por lo que sabemos de antemano que el resultado es ϕ.