

Teoría de Lenguajes
Teoría de la Programación I
Soluciones

Ejercicio 1

Indique si las siguientes afirmaciones son verdaderas o falsas. Justifique en cada caso.

- a) El lenguaje generado por la siguiente gramática $G=(V,T,P,S)$ es recursivamente enumerable pero NO libre de contexto. Siendo: $T = \{a,b\}$ $V = \{S, S', C, F, B, X, D\}$
 $P = \{ S \rightarrow CS'F ; S' \rightarrow BS' \mid S'a \mid X; Xa \rightarrow aX; XF \rightarrow Da; Ba \rightarrow aB; BD \rightarrow Db; Ca \rightarrow aC; CD \rightarrow b \}$

FALSO.

La idea es analizar las derivaciones que surgen de las reglas de producción.

Tenemos $S \Rightarrow CS'F \Rightarrow CBS'F \Rightarrow CBBS'F \Rightarrow CBBS'aF \dots$ es decir, estas reglas generan o bien una **a** o bien una **B**

Las derivaciones terminan al final en la variable **X**

$S \Rightarrow CS'F \Rightarrow CBS'F \Rightarrow CBBS'F \Rightarrow CBBS'aF \Rightarrow CB...BXaa...aF$ quedando **CB^kXa^pF**

La **X** se "mueve" a la derecha y al encontrarse con la **F**, genera una **a** y una variable **D** ($XF \rightarrow Da$). Esto determina que las tiras terminan en **a**

Las **a**'s se "mueven" a la izquierda con la regla $Ba \rightarrow aB$

Las **B**'s a medida que se desplazan a la derecha, al encontrarse con la **D**, se "convierten" en **b**.

La **C**, saltea las **a**'s (si hubiera) y cuando se encuentre con la **D**, generan una **b** ($CD \rightarrow b$)

Lo que se genera entonces, son tiras sin ningún tipo de restricciones entre sus símbolos terminales que pueden ser representadas por la expresión regular **a*bb*a**; de donde, el lenguaje generado por la gramática es regular y por tanto libre de contexto (por la Jerarquía de Chomsky).

- b) Si L_a NO es regular y $L_a \cup L_b$ es regular, entonces L_b NO es regular.

FALSO. Alcanza con considerar L_b como Σ^* , siendo Σ el alfabeto de L_a

- b) Si L_c es un lenguaje libre de contexto, entonces L_c es infinito.

FALSO. Todo lenguaje regular es, además, un lenguaje libre de contexto. En particular, los lenguajes *finitos* (con un número finito de elementos) son regulares. Por lo tanto, basta considerar un lenguaje finito, para mostrar que no todo lenguaje libre de contexto necesariamente es infinito.

- c) Si L_d y L_e son libres de contexto pero NO regulares, $L_d \neq L_e$ y $L_d \cap L_e \neq \emptyset$, entonces $L_d \cap L_e$ es libre de contexto pero NO regular.

FALSO. Sea por ejemplo:

$L_d = \{ x / x \text{ es de la forma } a^k b^k, k \geq 0 \}$ que sabemos es libre de contexto y NO regular y

$L_e = \{ x / x \text{ es de la forma } c^p d^p, p \geq 0 \}$ que sabemos es libre de contexto y NO regular

Su intersección es $\{\epsilon\}$ que tiene solo una tira y por lo tanto es un lenguaje finito. Entonces es regular.

Ejercicio 2

Sea $L_2 = \{ \text{tiras de la forma } w\#ww, \text{ con } w \in \{a,b\}^* \}$ un lenguaje NO libre de contexto.

- a) Construya una gramática $G_2 / L_2 = L(G_2)$. Explique brevemente.

$S \rightarrow I S' F$

/* Genero por cada a una A y una C correspondiente a cada w respectivamente*/

/* Genero por cada b una B y una D correspondiente a cada w respectivamente*/

$S' \rightarrow aACS' \mid bBDS' \mid \#$

/* Muevo el # a la izquierda */

$A\# \rightarrow \#A$

$B\# \rightarrow \#B$

$C\# \rightarrow \#C$

$D\# \rightarrow \#D$

/* Muevo los terminales a la izquierda */

$Aa \rightarrow aA$

$Ab \rightarrow bA$

$Ba \rightarrow aB$

$Bb \rightarrow bB$

$Ca \rightarrow aC$

$Cb \rightarrow bC$

$Da \rightarrow aD$

$Db \rightarrow bD$

/* Muevo la marca I a la derecha */

$Ia \rightarrow aI$

$Ib \rightarrow bI$

/* Cambio la marca I por I' para garantizar que no convierto las variables en terminales antes de que pasen el numeral 8?

$I\# \rightarrow \#I'$

/* Muevo las As y Bs a la izquierda y las Cs y Ds a la derecha */

$CA \rightarrow AC$

$CB \rightarrow BC$

$DA \rightarrow AD$

$DB \rightarrow BD$

/* Convierto las A y B en a y b respectivamente cuando chocan con marca I' */

$I'A \rightarrow aI'$

$I'B \rightarrow bI'$

/* Convierto las C y D en a y b respectivamente cuando chocan con la marca F */

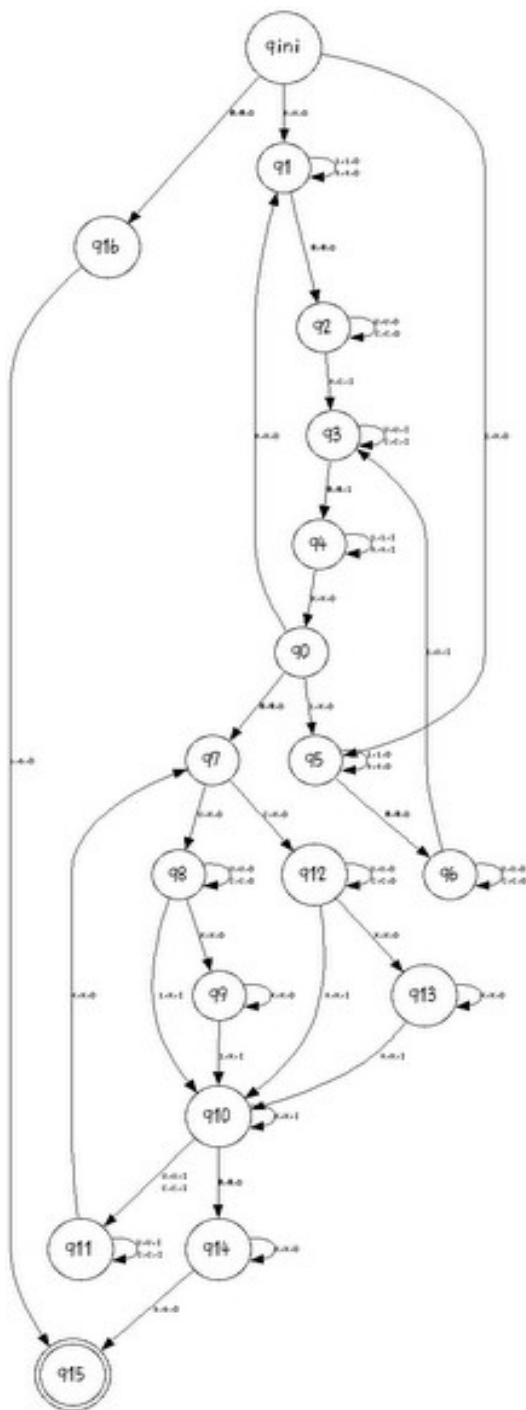
$CF \rightarrow Fa$

$DF \rightarrow Fb$

/* Elimino la marca asegurando que no quedan variables sin convertir */

$I'F \rightarrow \epsilon$

- b) Construya un autómata $M_2 / L_2 = L(M_2)$. Explique brevemente.



La idea de la máquina de Turing es separar $w\#ww$ en $w_1\#w_2w_3$ con $w_1 = w_2 = w_3$ y chequear inicialmente que $w_1 = w_2$ y luego chequear que $w_2 = w_3$.

Por lo tanto la solución parece "repetirse" dos veces, con la única diferencia que en la primera (para chequear $w_1 = w_2$) se tiene un numeral entre medio de las tiras.

Por último una vez que se tiene todo marcado avanzamos hacia el final de la tira hasta encontrar un blanco para asegurar que no existe ningún elemento sin marcar.

Ejercicio 3

Sea $L_3 = \{w / w \text{ es de la forma } b^p a^k b^r \text{ con } r = \lfloor k/3 \rfloor + 1, k \geq 0, p > 0\}$

a) Clasifique a L_3 según la Jerarquía de Chomsky.

L_3 es libre de contexto pero no regular.

En primer lugar se prueba que el lenguaje no es regular utilizando el contra-recíproco del Pumping Lema para Lenguajes Regulares.

Sea $N \in \mathbb{N}$ la constante del PL, se elige la tira z tal que $z = ba^{3N}b^{N+1}$.

Se cumple que $z \in L_3 \wedge |z| = 4N+2 > N$

Ahora bien, todas las descomposiciones uvw de z tales que $|uv| \leq N \wedge |v| \geq 1$ son:

Familias	b	a^{3N}	b^{N+1}
1	v	v	
2		v	

Para cada una de las familias de descomposiciones anteriores se debe encontrar un $i \in \mathbb{N}$ tal que $uv^i w \notin L_3$.

Familia 1:

$u = \varepsilon$

$v = ba^j$ $j \geq 0$

$w = a^{3N-j}b^{N+1}$

Se tiene que $uv^i w = (ba^j)^i a^{3N-j}b^{N+1}$

Eligiendo $i=0$, $z_0 = a^{3N-j}b^{N+1} \notin L_3$ pues la tira no comienza con b.

Familia 2:

$u = ba^j$

$v = a^k$

$w = a^{3N-j-k}b^{N+1}$ $j+k+1 \leq N \wedge k \geq 1$

Se tiene que $uv^i w = ba^j(a^k)^i a^{3N-j-k}b^{N+1} = ba^{3N+(i-1)k}b^{N+1}$

Eligiendo $i=4$ dado que $k \geq 1$ se tiene que $3N+3k \geq 3N+3$

Por tanto,

$$\lfloor (3N+3k)/3 \rfloor + 1 \geq \lfloor (3N+3)/3 \rfloor + 1 = \lfloor N+1 \rfloor + 1 = N+2 > N+1$$

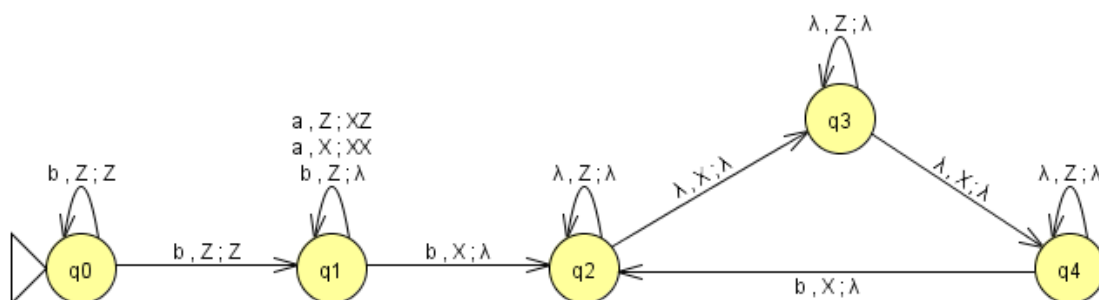
La cantidad de **b**'s es menor que la cantidad de **a**'s que quedan (para cualquier $k \geq 1$) según la relación definida. Entonces $z_4 \notin L_3$

Dado que se ha probado que para toda descomposición posible de z (que verifica las condiciones $|uv| \leq N \wedge |v| \geq 1$) existe un $i \in \mathbb{N}$ tal que $uv^i w \notin L_3$, en virtud del contra-recíproco del PL se puede afirmar que L_3 no es regular.

Para probar que L_3 es libre de contexto alcanza con construir un APD (parte (b)) o una GLC (parte (c))

b) Construya un autómata $M_3 / L_3 = L(M_3)$. Indique si su autómata es o no determinístico. Justifique adecuadamente.

$M_3 : (\{q_0, q_1, q_2, q_3, q_4\}, \{a, b\}, \{X, Z\}, \delta, q_0, Z)$



Nota: en el diagrama el símbolo λ representa ϵ

El APD es no determinista ya que $\delta(q_0, b, Z) = \{(q_0, Z), (q_1, Z)\}$, por lo que no se cumple la condición: $|\delta(q_i, a, z)| \leq 1, \forall a \in \Sigma \cup \{\epsilon\}, z \in \Gamma, q_i \in Q$.

c) Construya una gramática $G_3 / L_3 = L(G_3)$. ¿Está simplificada? Justifique.

$G_3 : (\{A, S\}, \{a, b\}, P, S)$

$P = \{$

$S \rightarrow bS \mid bAb,$

$A \rightarrow aaaAb \mid aa \mid a \mid \epsilon$

$\}$

La gramática presentada NO está simplificada porque tiene producciones épsilon.

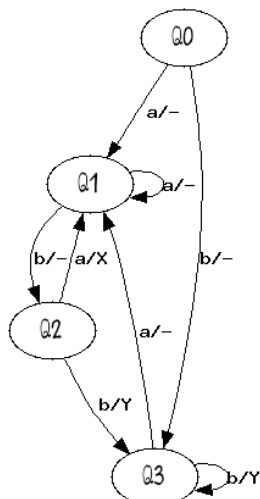
Ejercicio 4 - Teoría de Lenguajes

a) Construya una máquina de Mealy $M: (Q, \Sigma, \Lambda, \delta, \lambda, q_0)$, con $\Lambda = \{X, Y\}$ y $\Sigma = \{a, b\}$ que imprime:

- Una X por cada secuencia **aba**
- Una Y por cada secuencia **bb**

Ejemplos:

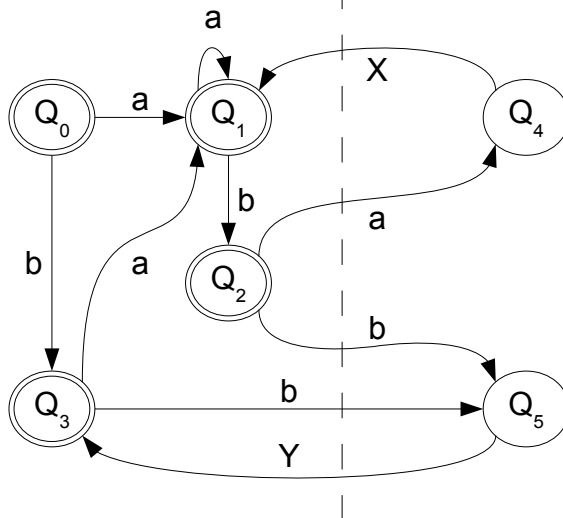
Entrada	Salida
aabb	Y
bababbababaaab	XYXX
ϵ	ϵ
bbaaaabbbaba	YYYX
baaaaab	ϵ



Nota: Las gramáticas y los autómatas deben corresponderse con el tipo del lenguaje considerado en cada caso, según la Jerarquía de Chomsky. Se valora positivamente la simplicidad de las soluciones propuestas así como una breve explicación de éstas. Todas las respuestas deben estar debidamente justificadas.

b) Construya un Autómata Finito Determinista de dos cintas que acepte

$\{ (w, f(w)) \mid f(w) \text{ la función que genera la salida de la máquina de la parte (a)} \}$



¿Se puede construir un AFD de 2 cintas, que reconozca el conjunto de pares tiras descrito en la parte anterior, comenzando a leer desde cualquier cinta? Justifique su respuesta.

NO. Solo se puede comenzar leyendo de la cinta izquierda. Esto se demuestra con el AFD de dos cintas construido en la parte anterior (que comienza por la cinta izquierda), y observando, por ejemplo, que para el par de tiras $\langle a, \epsilon \rangle$ no se puede comenzar a leer por la cinta derecha, porque no tenemos un símbolo para consumir y pasar a un estado donde se pueda consumir el símbolo "a" de la tira de la izquierda.