Autómatas y Lenguajes Formales
Tarea 2

Rubí Rojas Tania Michelle

11 de octubre de 2018

1. Muestra que los lenguajes libres del contexto no son cerrados bajo intersección ni complemento.

   Demostración. Probaremos que los lenguajes libres del contexto no son cerrados bajo la operación intersección usando un contraejemplo. Sean $L_1 = \{a^n b^n c^m : m, n \geq 0\}$ y $L_2 = \{a^m b^n c^n : m, n \geq 0\}$ lenguajes libres del contexto. Entonces $L_1 \cap L_2 = \{a^n b^n c^n : n \geq 0\}$, el cual no es un lenguaje libre del contexto, por lo visto en clase. Por lo tanto, los lenguajes libres del contexto no son cerrados bajo la intersección.
   Ahora, probaremos que los lenguajes libres del contexto no son cerrados bajo la operación complemento. Si el complemento de un lenguaje libre del contexto fuera siempre libre del contexto entonces podríamos demostrar la cerradura de la intersección usando las leyes de De Morgan:

   $$L_1 \cap L_2 = (L_1^c \cup L_2^c)^c$$

   Pero como los lenguajes libres del contexto no son cerrados bajo la intersección, entonces tampoco pueden ser cerrados bajo complemento.Por lo tanto, los lenguajes libres del contexto no son cerrados bajo el complemento.

   □

2. Demuestra que si M es un AFD, existe un APN que simula a M aceptando por estado final y sin remover símbolos de la pila durante su ejecución.

   Demostración. Supongamos que M es un AFD. Lo que debemos hacer es que un M_P, que es un APN, imite el comportamiento de M. Si M_P es un autómata de pila que acepta por estado final y no remueve ningún símbolo de la pila, entonces claramente M_P se comporta como un AFD.
   Formalmente, sea $M = (Q, \Sigma, \delta, q_0, F)$ un autómata finito determinista. Construimos
   $M_A = (Q, \Sigma, \varnothing, \delta', q_0, F)$ un autómata de pila, con $\delta' = \{((p, u, \epsilon), (q, \epsilon)) : (p, u, q) \in \delta\}$ que acepta el mismo lenguaje. Es claro que con esta transición, el autómata de pila reconoce el mismo lenguaje que el AFD. Por lo tanto, M_P simula a M aceptando por estado final y sin remover símbolos.

   □

3. Da gramáticas libres del contexto que generen los siguientes lenguajes sobre el alfabeto $\sum = \{a, b\}$:

- $\{w \mid w$ inicia y termina con el mismo símbolo$\}$
Solución: La gramática que genera al lenguaje es la siguiente:

$$S \to a \mid b \mid aTa \mid bTb$$
$$T \to \epsilon \mid aT \mid bT$$

- $\{w \mid w$ es de longitud impar y el símbolo de enmedio es $b\}$
Solución: La gramática que genera al lenguaje es la siguiente:

$$S \to aSa \mid bSb \mid aSb \mid bSa \mid b$$

- $\{w \mid w$ tiene menos símbolos a que símbolos $b\}$
Solución: La gramática que genera al lenguajes es la siguiente:

$$S \to bS \mid bT \mid TS$$
$$T \to \epsilon \mid aTb \mid bTa \mid TT$$

- $\{w \mid w$ tiene tantos símbolos a como símbolos $b\}$
Solución: La gramática que genera al lenguajes es la siguiente:

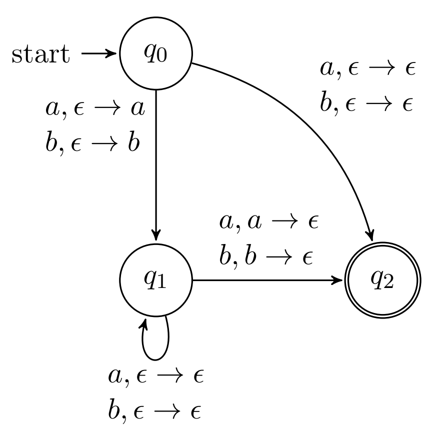
$$S \to \epsilon \mid aSb \mid bSa \mid SS$$

- $\{ww^R \mid w \in \sum^*\}$
Solución: La gramática que genera al lenguajes es la siguiente:

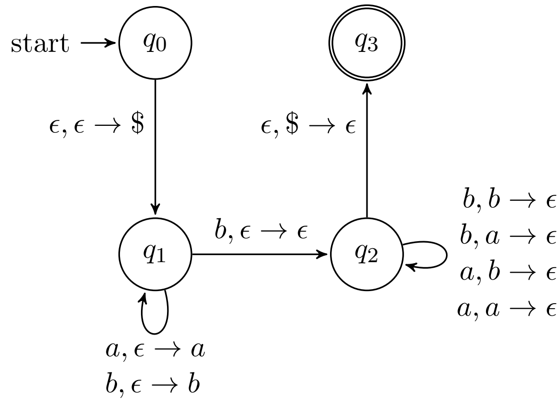
$$S \to \epsilon \mid aSa \mid bSb$$

4. Elige tres lenguajes del inciso anterior y construye autómatas de pila que los reconozcan.

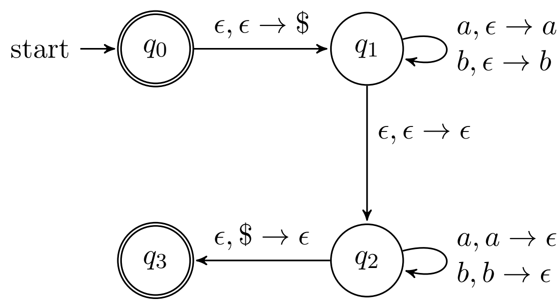
- $\{w \mid w$ inicia y termina con el mismo símbolo$\}$
Solución:



- $\{w \mid w$ es de longitud impar y el símbolo de enmedio es $b\}$
  Solución:



- $\{ww^R \mid w \in \sum^*\}$
  Solución:



5. El lenguaje *Tak!* se define de la siguiente manera:

$$\text{T} \to \text{Tak!} \mid \langle T \rangle \mid \text{TT}$$

Construye un autómata de pila sobre el alfabeto $\sum = \{\langle, \rangle, T, a, k, !\}$ que lo reconozca y describe informalmente la ejecución con la cadena $\langle Tak!Tak! \rangle$.
Solución: Las transiciones del autómata de pila correspondiente están dadas en la tabla siguiente:

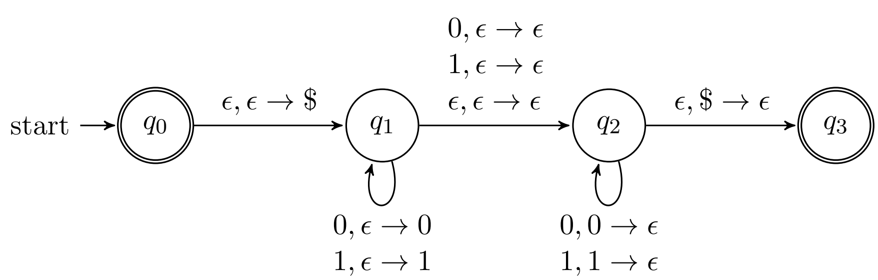
1	$(q_0, \epsilon, \epsilon)$	(q_1, T)
2	(q_1, ϵ, T)	$(q_1, Tak!)$
3	(q_1, ϵ, T)	$(q_1, \langle T \rangle)$
4	(q_1, ϵ, T)	(q_1, TT)
5	$(q_1, Tak!, Tak!)$	(q_1, ϵ)
8	$(q_1, \langle, \langle)$	(q_1, ϵ)
9	$(q_1, \rangle, \rangle)$	(q_1, ϵ)

El funcionamiento de este autómata de pila ante la cadena $\langle Tak!Tak!\rangle$ aparece en la siguiente tabla:

Estado	Falta leer	Pila
q_0	$\langle Tak!Tak!\rangle$	ϵ
q_1	$\langle Tak!Tak!\rangle$	T
q_1	$\langle Tak!Tak!\rangle$	$\langle TT\rangle$
q_1	$Tak!Tak!\rangle$	$TT\rangle$
q_1	$Tak!Tak!\rangle$	$Tak!T\rangle$
q_1	$Tak!\rangle$	$T\rangle$
q_1	$Tak!\rangle$	$Tak!\rangle$
q_1	$\rangle$	$\rangle$
q_1	ϵ	ϵ

6. Considera el lenguaje $L = \{w \in \{0,1\}^* \mid w^R = \overline{w}\}$ donde $\overline{x}$ es la cadena x con cada uno de sus bits invertidos, por ejemplo $\overline{0101} = 1010$.

   - Da un autómata de pila que reconozca L.
     Solución:



   - Da una gramática libre del contexto que genere L.
     Solución: La gramática que genera al lenguaje es la siguiente:

$$S \to 0 \mid 1 \mid 0S0 \mid 1S1 \mid \epsilon$$

7. Elige un lenguaje del inciso 3 y da una gramática en forma normal de Chomsky que lo genere. Además, da una gramática en forma normal de Griebach que lo genere.
   Solución: Sea $L = \{w \mid w$ es de longitud impar y el símbolo de enmedio es $b\}$ el lenguaje generado por la siguiente gramática:

$$S \to aSa \mid bSb \mid aSb \mid bSa \mid b$$

La gramática en Forma Normal de Chomsky que lo genera es:

$$\begin{gathered} S \to TX \mid UY \mid TY \mid UX \mid U \\ T \to a \\ U \to b \\ X \to ST \\ Y \to SU \end{gathered}$$

La gramática en Forma Normal de Griebach que lo genera es:

$$
\begin{gathered}
S \to aX \mid bY \mid aY \mid bX \mid b \\
T \to a \\
U \to b \\
X \to aXT \mid bYT \mid aYT \mid bXT \mid bT \\
Y \to aXU \mid bYU \mid aYU \mid bXU \mid bU
\end{gathered}
$$

8. Muestra que los siguientes lenguajes no son libres del contexto:

   - $\{a^p \mid p \text{ es un número primo}\}$

     Demostración. Veamos que L no es un lenguaje libre del contexto.

     i) Sea $k \in \mathbb{N}$.
     ii) $z = a^k$, donde k es un número primo que es al menos $n+2$, donde n es la longitud del bombeo.
     iii) $z = uvwxy$, $|vx| > 0$, $|vwx| \leq k$.
     iv) Sea $i = n$. Si $|vx| = m$ y $z' = uv^{k-m}wx^{k-m}y$, entonces $|z'| = m(k-m)+m = (m+1)(k-m)$. Tenemos que $m+1$ debe ser al menos 2, por lo que $|vx| > 0$. Además, $k-m$ debe ser al menos 2 ya que k es al menos $n+2$, y $|vx| \leq |vwx| \leq n$. Por lo anterior, $|z'|$ no puede ser un número primo.

     Por lo tanto, L no es un lenguaje libre del contexto.

     □

   - $\{a^{2^n} \mid n \in \mathbb{N}\}$

     Demostración. Veamos que L no es un lenguaje libre del contexto.

     i) Sea $k \in \mathbb{N}$.
     ii) $z = a^{2^k}$.
     iii) $z = uvwxy$, $|vx| > 0$, $|vwx| \leq k$.
     iv) Sea $i = 2$. Si $v = a^{2^n}$ y $x = a^{2^m}$ entonces sabemos que $2^n + 2^m = |vx| \leq |vwx| \leq k$ y $2^n + 2^m = |vx| > 0$ por lo que $0 < 2^n + 2^m \leq k$.
     Notemos que $|uv^iwx^iy| = 2^k + (2^n + 2^m) \leq 2^k + k < 2^k + 2^k = 2^{k+1}$ siempre y cuando $k > 0$, así al bombear la cadena uv^iwx^iy, ésta tiene estrictamente una longitud entre 2^k y 2^{k+1}. Ninguna cadena de este tipo puede estar en el lenguaje, por lo que $uv^iwx^iy \notin L$.

     Por lo tanto, L no es un lenguaje libre del contexto.

     □

   - $\{w \in \{a,b,c\}^* \mid n_a(w) = n_b(w) = n_c(w)\}$

     Demostración. Veamos que L no es un lenguaje libre del contexto.
     Si $L = \{w \in \{a,b,c\}^* \mid n_a(w) = n_b(w) = n_c(w)\}$ entonces $L \cap \{a^*b^*c^*\} = \{a^nb^nc^n\}$. Sabemos que la intersección entre un lenguaje libre del contexto (L) y un lenguaje regular ($a^*b^*c^*$) es un lenguaje libre del contexto. Pero sabemos que $\{a^nb^nc^n\}$ no es un lenguaje libre del contexto, contradicción. Por lo tanto, L no es un lenguaje libre del contexto.

     □

- $\{0^n1^n0^n1^n \in \{0,1\}^* \mid n \in \mathbb{N}\}$

  Demostración. Veamos que L no es un lenguaje libre del contexto.

  i) Sea $k \in \mathbb{N}$.
  ii) $z = 0^k1^k0^k1^k$.
  iii) $z = uvwxy$, $|vx| > 0$, $|vwx| \leq k$.
  iv) Entonces vwx puede ser como los siguientes casos:
     - $v \in 0^k$ y $x \in 0^k$. Esto no es posible porque se infrige la regla de que $|vwx| \leq k$, ya que w consistiría de al menos k números. Por lo tanto, $z \notin L$.
     - $v \in 1^k$ y $x \in 1^k$. Esto no es posible porque se infrige la regla de que $|vwx| \leq k$, ya que w consistiría de al menos k números. Por lo tanto, $z \notin L$.
     - $v \in 0^k$ y $x \in 1^k$. Al hacer $i = 0$, dado que $|vx| > 0$, a^k y/o en 1^k tendríamos menos números que en 0^k y/o 1^k, por lo que $z_0 =\notin L$.
     - v o x ocupan posiciones sobre la frontera entre dos regiones. Esto es, cualquiera de las dos (pero no ambas) consiste de $a...ab...b$; al bombear la cadena para $i = 2$, se mezclarían las $a$es y $b$es y no se respetarían las regiones, por lo que el resultado de bombear no estaría en L.

  Por lo tanto, L no es un lenguaje libre del contexto.

  □

9. Ejecuta detalladamente el algoritmo CYK con la gramática:

$$S \to AB$$
$$A \to a$$
$$B \to AB \mid b$$

y con la cadena $w = abbb$.

Solución: Dada la cadena $w = abbb$ y la gramática G, veamos si $w \in L(G)$.
Sea $|_0\ a\ |_1\ b\ |_2\ b\ |_3\ b\ |_4$ la cadena w dividida en subíndices, esto para tener una ayuda visual para ejecutar el algoritmo.

i) Inicializamos la tabla con la longitud de la cadena, $n = |w| = 4$. Comenzamos con las subcadenas de longitud 1. Sea $C = X_{i,i+1}$. Si existe una producción $A \to C$, entonces escribimos A en la entrada $i, i+1$ de la tabla.

0				
A	1			
	B	2		
		B	3	
			B	4

ii) Seguimos con las producciones para subcadenas de longitud dos. La subcadena $X_{i,i+2}$ se divide en dos cadenas: $X_{i,i+i}$ y $X_{i+1,i+2}$. Si hay una producción $Z \to XY$ tal que $X = (i, i+1)$ y $Y = (i+1, i+2)$ entonces colocamos Z en la entrada (i, i+2).

0				
A	1			
S, B	B	2		
	∅	B	3	
		∅	B	4

iii) El siguiente paso son las producciones para las subcadenas de longitud tres. La subcadena $X_{i,i+3}$ sigue instrucciones análogas al paso dos. Tenemos sólo dos casos: $X_{0,3} = X_{0,1}X_{1,3} = X_{0,2}X_{2,3}$ y $X_{1,4} = X_{1,2}X_{2,4} = X_{1,3}X_{3,4}$.

0				
A	1			
S, B	B	2		
$\varnothing$	$\varnothing$	B	3	
	$\varnothing$	$\varnothing$	B	4

iv) Concluimos con las producciones para las sucadenas de longitud cuatro. La subcadena $X_{i,i+4}$ sigue las instrucciones análogas al paso dos. Tenemos únicamente un caso:
$X_{0,4} = X_{0,3}X_{3,4} = X_{0,1}X_{1,4}$

0				
A	1			
S, B	B	2		
$\varnothing$	$\varnothing$	B	3	
$\varnothing$	$\varnothing$	$\varnothing$	B	4

Como el símbolo inicial no aparece en la entrada (a,n) entonces podemos concluir que $w \notin L(G)$.