

Teoría de Lenguajes
Solución 2do. Parcial – Curso 2013

Ejercicio 1 [Evaluación individual del obligatorio]

- a)
iv. Cuando se realiza un “reduce”
- b)
ii. La gramática implementada en el archivo Sintactico.sin no se corresponde con el lenguaje especificado en la letra del obligatorio.
- c)
i. Construyen el árbol de derivación en forma ascendente (de las hojas hacia la raíz).
- d)
iii. No es correcta porque no reconoce el mismo lenguaje
- iv. y v.** No son correctas porque JCUP no puede generar el parser a partir de una gramática ambigua. Y por ejemplo para la tira 1+2+3 estas gramáticas tiene más de una árbol de derivación posible.

Ejercicio 2

Dado el siguiente lenguaje:

$$L_2 = \{ a^p b^q c^r, r > q \text{ o } q > p; \quad p, q, r > 0 \}$$

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

La idea es dividir en dos casos, uno cuando $r > q$ y otro cuando $q > p$

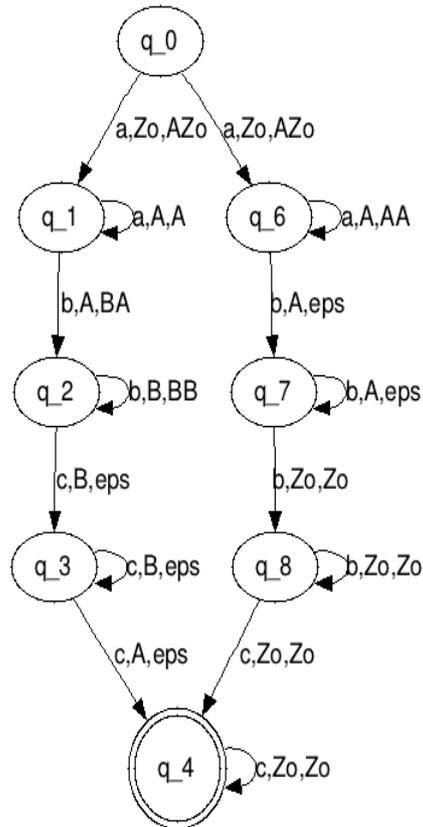
$S \rightarrow S_1 \mid S_2$
 $S_1 \rightarrow ABC$
 $A \rightarrow a \mid aA$
 $B \rightarrow bc \mid bBc$
 $C \rightarrow c \mid cC$
 $S_2 \rightarrow DEC$
 $D \rightarrow ab \mid aDb$
 $E \rightarrow b \mid bE$

La gramática no está simplificada, porque tiene producciones unitarias, las eliminamos (S_1 y S_2 ya no son útiles, los eliminamos)

$S \rightarrow ABC \mid DEC$
 $A \rightarrow a \mid aA$
 $B \rightarrow bc \mid bBc$
 $C \rightarrow c \mid cC$
 $D \rightarrow ab \mid aDb$
 $E \rightarrow b \mid bE$

ya está simplificada: no tiene producciones épsilon ni unitarias, todos los símbolos son útiles (se verifica realizando los algoritmos para variables positivas y luego alcanzables).

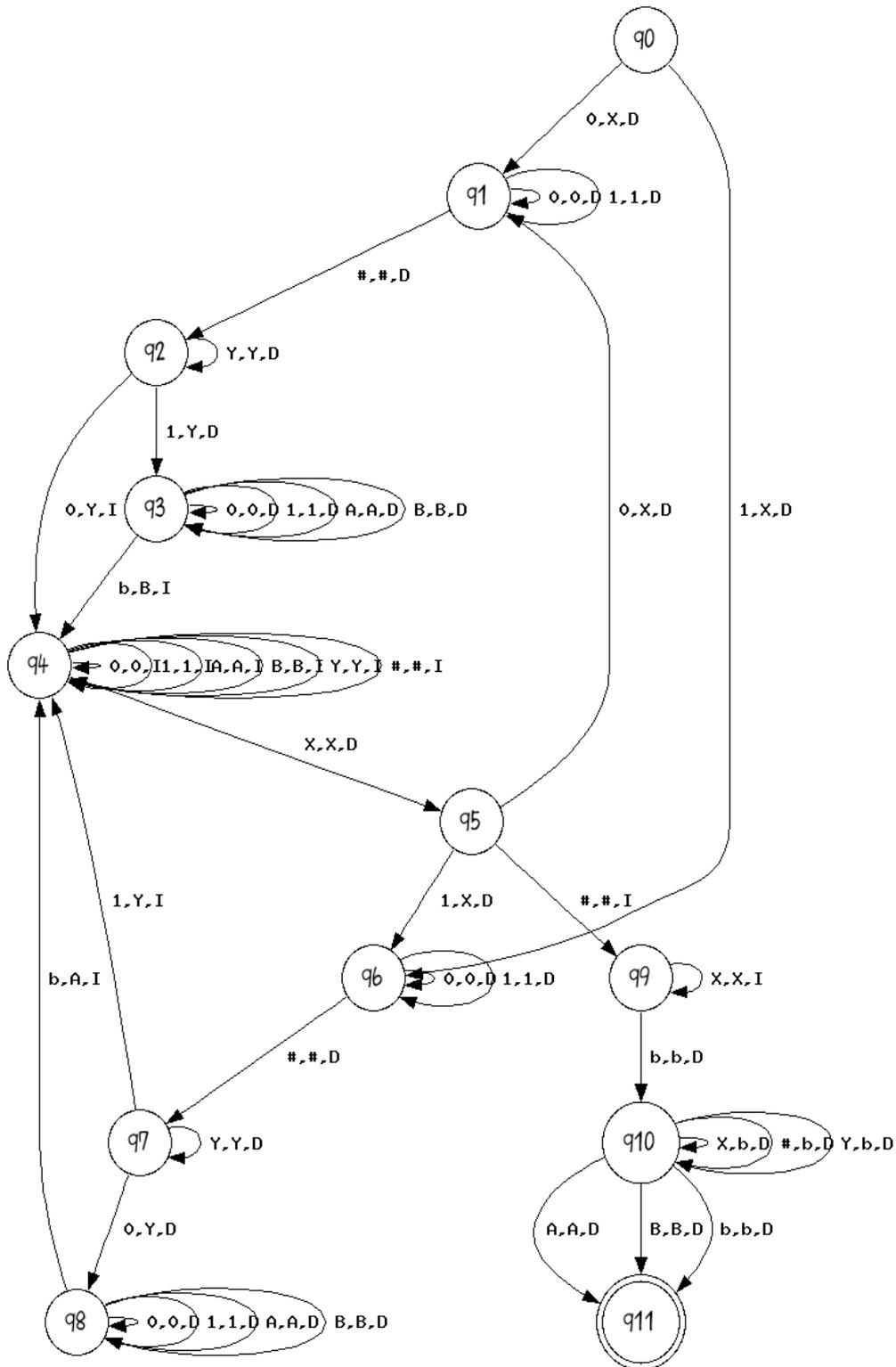
b) Construya un autómata que reconozca a L_2 . ¿Es determinista? Justifique.



El autómata no es determinista: a partir del estado q_0 hay dos transiciones hacia estados diferentes etiquetadas con el mismo símbolo de entrada, y con el mismo símbolo en el tope del stack.

Ejercicio 3

a)



Nota: Se valora positivamente la simplicidad de las soluciones propuestas así como una breve explicación de éstas. Todas las respuestas deben estar debidamente justificadas.

b) $L_3 = \{ x\#y\#z, \text{ donde } x \text{ es la secuencia de jugadas del jugador A, } y \text{ es la secuencia de jugadas del jugador B, y } z \text{ es una tira de A's y/o B's, cuyo largo es la cantidad de manos ganadas (por uno u otro jugador)} \}$

Demostrar que L_3 no es libre de contexto

L_3 es Recursivamente Enumerable (lo cual demostraremos en la parte b, construyendo la MT) y no es Libre de Contexto (lo cual demostraremos por el contra recíproco del PL para L.C.)

Sea N la cte. del PL, elijo $z = 0^N \# 1^N \# B^N = u.v.w.x.y$

Casos	0...0	#	1...1	#	B...B
1	v x				
2	v x...x		
3	v		x		
4	v...v x		
5			v x		
6			v x...x
7			v		x
8			v...v x
9					v x

Caso 1

$$\begin{aligned}
 u &= 0^j \\
 v &= 0^k \\
 w &= 0^l \\
 x &= 0^p \\
 y &= 0^{N-j-k-l-p} \# 1^N \# B^N
 \end{aligned}
 \qquad
 \begin{aligned}
 k+l+p &\leq N \\
 k+p &> 0
 \end{aligned}$$

$z_i = 0^{N+(i-1)(k+p)} \# 1^N \# B^N$, con $i=2$, $z_2 = 0^{N+k+p} \# 1^N \# B^N$, como $k+p > 0$, la cantidad de jugadas del primer jugador tiene más símbolos que la cantidad de jugadas del segundo jugador.

Los casos 5 y 9 son análogos. La diferencia está en el caso 9, donde lo que se desbalancea es la cantidad de manos ganadas por el segundo jugador es mayor que la cantidad de jugadas.

Caso 2

$$\begin{aligned}
 u &= 0^{N-k-l-p} \\
 v &= 0^k \\
 w &= 0^l \\
 x &= 0^p \# 1^q \\
 y &= 1^{N-q} \# B^N
 \end{aligned}
 \qquad
 k+l+p+q < N$$

$z_i = 0^{N+(i-1)k-p} (0^p \# 1^q)^i 1^{N-q} \# B^N$, con $i=0$, $z_0 \notin L_3$ por tener solamente un #.

Los casos 4, 6 y 8 son análogos.

Caso 3

$$\begin{aligned}
 u &= 0^{N-k-l} \\
 v &= 0^k \\
 w &= 0^l \# 1^p \\
 x &= 1^q \\
 y &= 1^{N-p-q} \# B^N
 \end{aligned}
 \qquad
 \begin{aligned}
 k+l+p+q &< N \\
 k+q &> 0
 \end{aligned}$$

$z_i = 0^{N+(i-1)k} \# 1^{N+(i-1)q} \# B^N$, con $i=0$, $z_0 = 0^{N-k} \# 1^{N-q} \# B^N$. Como $k+q > 0$, al menos uno de los índices debe ser mayor que cero. Por ende, la cantidad de triunfos va a

ser diferente por lo menos en la cantidad de jugadas del primero, del segundo o de ambos y la tira no está en L_3 .

El caso 7 es análogo, pero el argumento es que la cantidad de jugadas del primero o del segundo difieren de la cantidad de jugadas representadas por los triunfos del segundo jugador.

Como estas son todas las descomposiciones que cumplen $|vwx| \leq N$ y $|vx| > 0$, y para cada uno de ellos, existe un i tal que z_i no está en el lenguaje, **entonces L_3 no es libre de contexto.**

c) Construya una gramática G_3 tal que $L_3 = L(G_3)$.

$S \rightarrow T##$

$T \rightarrow TP \mid P$ - genero las jugadas

$P \rightarrow 1U \mid 0C \mid$
 $1CA_1 \mid 0UB_1$ - genero los empates y las jugadas ganadas por los jugadores A y B

$U1 \rightarrow 1U$

$U0 \rightarrow 0U$

$C1 \rightarrow 1C$

$C0 \rightarrow 0C$

- se desplazan a la derecha las jugadas del jugador B

$U\# \rightarrow \#1$

$C\# \rightarrow \#0$

- al pasar por el primer $\#$ se cambian por **0** ó **1**, según el caso

$A_10 \rightarrow 0A_1$

$A_11 \rightarrow 1A_1$

$B_10 \rightarrow 0B_1$

$B_11 \rightarrow 1B_1$

- se desplaza a la derecha el indicador de quien es el jugador que ganó una jugada determinada

$A_1\# \rightarrow \#A_2$

$B_1\# \rightarrow \#B_2$

- al encontrarse con el primer $\#$ se cambia de variable

$A_20 \rightarrow 0A_2$

$A_21 \rightarrow 1A_2$

$B_20 \rightarrow 0B_2$

$B_21 \rightarrow 1B_2$

- se sigue desplazando a la derecha el indicador de quien es el jugador que ganó una jugada determinada

$A_2\# \rightarrow \#A$

$B_2\# \rightarrow \#B$

- se transforman las variables **A_2** y **B_2** en los terminales **A** y **B** según sea quien ganó la jugada al encontrar el segundo $\#$