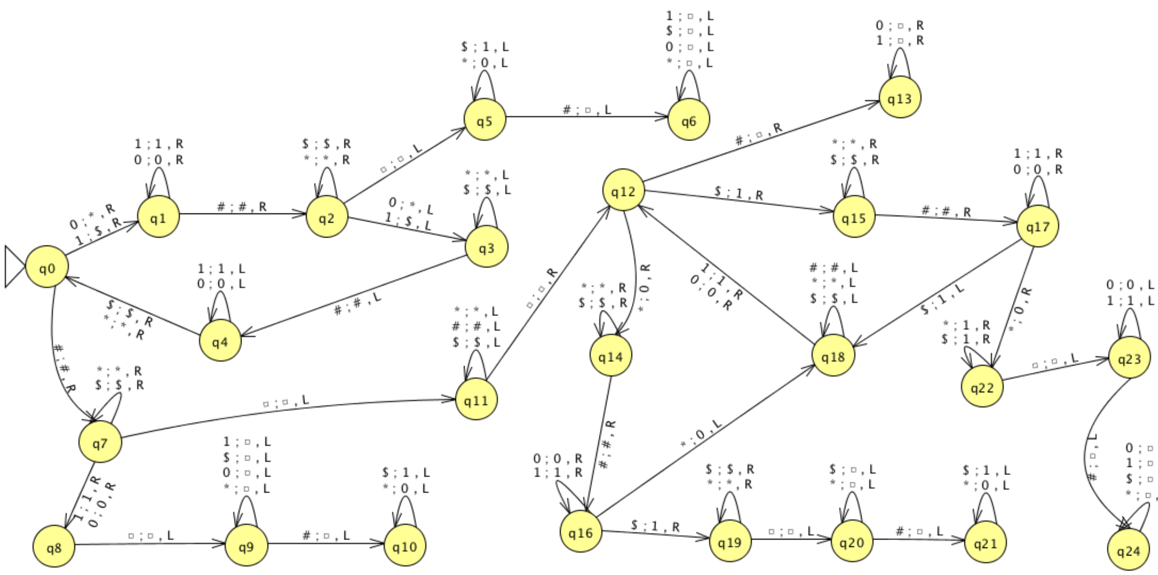
Universidad Técnica Federico Santa María
Campus Santiago San Joaquín
Segundo Semestre 2016

Tarea 4 - TALF

Jorge Contreras 201573547-6
Juan Pablo Jorquera 201573533-6

1. Pregunta 1



La máquina de Turing se divide en dos grandes partes: inicialmente compara los tamaños para decidir cual es el menor y luego, de ser iguales compara término a término para definir el menor.

- Entre q0 y q4 va marcando los 1 con $ y los 0 con y luego verifica que exista ese dígito en el término de la derecha.
- En q2 se sale al haber encontrado un dígito a la izquierda y haber terminado por completo el de la derecha, entonces el de la derecha es más corto, y en q5 y q6 lo decodifica y limpia el resto.
- En q0 se sale al haber terminado el primer término, entonces avanza hasta la posición actual del término de la derecha y verifica si también terminó.
- En caso de no haber terminado significa que el término de la izquierda era menor, entonces en q8, q9 y q10, limpia el término de la derecha y decodifica el de la izquierda.
- Por otro lado si también termina el término de la derecha, significa que eran de igual tamaño, donde avanza a q11 y de aquí se devuelve al comienzo de la cinta.
- Acá para comparar son dos partes virtualmente simétricas, dependiendo si comienza con 0 o 1 (codificado), lo decodifica y avanza al término de la derecha para comparar hasta q16 y q17 respectivamente.

- En caso de encontrar el mismo valor, se devuelve a q12 para seguir comparando.
- De no ser así se sale en q19 y q22 para terminar devolviendo el término adecuado en el resto de los nodos de esta rama.
- Finalmente en la situación de que ambos sean completamente iguales, se sale de q12 a q13, estándo ya decodificados por el loop los términos, simplemente borra el de la derecha.

2. Pregunta 2

S → bS | Sa | aSb | ε

(a) Primero, aseamos la gramática

S_o→ S | ε

S → bS | Sa | aSb | b | a | ab

Luego, lo dejamos en FNC

S_o→ S | ε

S → BS | SA | XB | b | a | AB

A → a

B → b

X → AS

(b) Palabra: aaa

a	a	a
↓	↓	↓
S,A	S,A	S,A

aa	aa
↓	↓
S,X	S,X

aaa

↙ ↘

a/aa	aa/a
↓	↓
X	S

aaa es formado por S y X, y por lo tanto pertenece a la gramática.

Palabra: bbaa

b	b	a	a
↓	↓	↓	↓
S,B	S,B	S,A	S,A

bb	ba	aa
↓	↓	↓
S	S	S,X

bba		baa	
↙	↘	↙	↘
b/ba	bb/a	b/aa	ba/a
↓	↓	↓	↓
S	S	S	S

	bbaa	
↙	↓	↘
b/baa	bb/aa	bba/a
↓	↓	↓
S	∅	S

bbaa es formado por S, por lo tanto pertenece a la gramática.

Palabra: aabba

a	a	b	b	a
↓	↓	↓	↓	↓
S,A	S,A	S,B	S,B	S,A

aa	ab	bb	ba
↓	↓	↓	↓
S,X	S,X	S	S

aab		abb		bba	
↙	↘	↙	↘	↙	↘
a/ab	aa/b	a/bb	ab/b	b/ba	bb/a
↓	↓	↓	↓	↓	↓
X	S	X	S	S	S

	aabb			abba	
↙	↓	↘	↙	↓	↘
a/abb	aa/bb	aab/b	a/bba	ab/ba	abb/a
↓	↓	↓	↓	↓	↓

X ∅ S X ∅ S

$$\begin{array}{cccc} & & \text{aabba} & \\ \swarrow & \swarrow & \searrow & \searrow \\ \text{a/abba} & \text{aa/bba} & \text{aab/ba} & \text{aabb/a} \\ \downarrow & \downarrow & \downarrow & \downarrow \\ \text{X} & \emptyset & \emptyset & \text{S} \end{array}$$

aabba es formado por S y X, por lo tanto pertenece a la gramática.

(c) El lenguaje acepta la palabra vacía y cualquier otra combinación que se pueda formar con las letras a y b.

3. Pregunta 3

$WHAT$!?!?!?

4. Pregunta 4

4.1. Parte a

Elijamos $w = a^{n^2} b^n c^n \ \epsilon L_1$, con $|vy| > 0$ y $|vxy| < n$
Sea w de la forma w = uvxyz, al bombear encontramos con 5 casos:
1) "u" e "y" se encuentren dentro de a^{n^2}.
2) "u" e "y" se encuentren dentro de b^n.
3) "u" e "y" se encuentren dentro de c^n.
4) "u" e "y" se encuentren en la intersección de a^{n^2} y b^n.
5) "u" e "y" se encuentren en la intersección de b^n y c^n.

Bombeamos por casos.

1) uvxyz se encuentran dentro de a^{n^2}.
$u = a^p$
$v = a^q$
$x = a^r$
$y = a^s$
$z = a^{n^2-p-q-r-s} b^n c^n$

k veces "v" e "y", quedando:
$a^{n^2+(q+s)(k-1)} b^n c^n$
eligiendo k=2
$a^{n^2+(q+s)} b^n c^n$

Para que pertenesca al lenguaje,

$n^2 + q + s =$ n*n
(q+s) = 0
Pero |vy|>0, existiendo así una contradicción.

2)Similar al caso 1, sólo que esta vez se bombea b^n, aumentando la cantidad de estas, quedando:
$a^{n^2}b^{n+k}c^n$
con cualquier k>1 nos escapamos del lenguaje.

3)Simétrico al caso 2, sólo que esta vez se bombea c^n.

4)"x" e "y" se encontrarán dentro de las a's y las b's, quedando algo del estilo:
$a^{n^2+k}b^{n+k}c^n$
bombeando hacia arriba con k=2 queda:
$a^{n^2+2}b^{n+2}c^n$
Como no se cumple la igualdad, nos escapamos del lenguaje.

5)"x" e "y" se encontrarán dentro de las b's y las c's, quedando algo del estilo:
$a^{n^2}b^{n+k}c^{n+k}$
bombeando hacia arriba con k=2 queda:
$a^{n^2}b^{n+2}c^{n+2}$
Como no se cumple la igualdad, nos escapamos del lenguaje.

Como nos escapamos en todos los casos posibles, L_1 no es LLC

4.2. Parte b

Primero vemos la ecuación de la forma $n = 8 - \frac{5}{3}m$, donde notamos que en los enteros no negativos solo hay dos combinaciones posibles:

- $m = 0$; $n = 8$
- $m = 3$; $n = 3$

Por lo que a continuación solo basta hacer el autómata correspondiente:

4b.png

Acá verificamos que incluso el lenguaje es regular, por lo que es de libre contexto.

4.3. Parte c

Al igual que antes, es más fácil armar el PDA viendo la ecución de otra forma: $5m = 24 + 3n$

4c.png

Como hay PDA que lo represente, entonces es LLC.

4.4. Parte d

A continuación se muestra el PDA del lenguaje, y como tiene PDA, entonces es LLC.

4d.png

4.5. Parte e

Elijamos $w = a^n b^n c a^n b^n \; \epsilon L_5$, con $|vy| > 0$ y $|vxy| < n$
Sea w de la forma w = uvxyz, al bombear nos encontramos con 7 casos:
1) "u" e "y" se encuentren dentro de las primera a^n.
2) "u" e "y" se encuentren dentro de las segundas a^n.
3) "u" e "y" se encuentren dentro de las primeras b^n.
4) "u" e "y" se encuentren dentro de las segundas b^n.
5) "u" e "y" se encuentren en la primera intersección de a's y b's.
6) "u" e "y" se encuentren en la segunda intersección de a's y b's.
7) "u" e "y" se encuentren en la intersección de las a^n con la c.

Bombeamos por casos.

1) uvxyz se encuentran dentro de a^n.
$u = a^p$
$v = a^q$
$x = a^r$
$y = a^s$
$z = a^{n-p-q-r-s}b^n ca^n b^n$

k veces "v" e "y", quedando:
$a^{n+(q+s)(k-1)}b^n ca^n b^n$
eligiendo k=0
$a^{n-q-s}b^n ca^n b^n$

Para que pertenesca al lenguaje,
n-q-s+n = n+n
-(q+s) = 0
Pero |vy|>0, existiendo así una contradicción.

2) Simétrico al caso 1, sólo que esta vez se bombea las segundas a^n.

3) Simétrico al caso 1, sólo que esta vez se bombean las primeras b^n.

4) Simétrico al caso 1, sólo que esta vez se bombean las segundas b^n.

5) |vxy| se encuentra entre $a^n b^n$, siguiendo la lógica del caso 1) es evidente que si bombeamos hacia arriba aumentamos los valores de a y b, y no se cumpliría la igualdad, ya que los primeros a's y b's son más que los segundos, escapando así del lenguaje.

6) Simétrico al caso 5, sólo que esta vez se bombean las segundas $a^n b^n$.

7) En los casos en que c se encuentre dentro de "v" o "y" es evidente que al bombear hacia arriba va a aumentar la cantidad de c en la palabra y se escapa logicamente del lenguaje, entonces el caso a analizar es cuando c se encuentra en x.

Así, si bombeamos hacia abajo, se eliminan las b's anteriores a c y las a's seguidas a c.
$a^n b^{n-k} ca^{n-k} b^n$
k<0 nos escapamos del lenguaje.

Como nos escapamos en todos los casos posibles, L_5 no es LLC

5. Pregunta 5

5.1. Parte a

- Convertir GLC G en autómata de pila y expresión regular R en AFD.
- Construir PDA que incluya a G y a R simultáneamente ($L(G) \cap L(R)$), aceptando sólo en caso de que ambos acepten.
- Transformar nuevo PDA a GLC.
- Aplicar CYK sobre dicho GLC para verificar la pertenencia.

5.2. Parte b

- Realizar aseo sobre GLC.
- En el GLC verificar si existe algún loop, es decir, si alguna variable se produce a sí misma, o bien se producen entre ellas. De ser así inmediatamente se sabe que existe una palabra w tal que $|w| \geq n$.
- Realizar árbol sobre la gramática con S en la raíz, poninendo en los hijos todas las opciones de variables.
- Repetir el paso anterior para cada hijo incluyendo las combinaciones posibles en cada una.
- En caso de encontrar alguno de los hijos con largo mayor a n, entonces dicha palabra existe. Por otro lado, si se terminó el árbol y no se encontró, entonces dicho lenguaje no contiene palabras con dichas características.