Teoría de Lenguajes

Consideraciones generales
i) Escriba nombre y C.I. en todas las hojas.
ii) Numere todas las hojas.
iii) En la primera hoja indique el total de hojas.
iv) Comience cada ejercicio en una hoja nueva.
v) Utilice las hojas de un solo lado.
vi) Entregue los ejercicios en orden.

Ejercicio 1 [8 puntos]

a) Indique si las siguientes afirmaciones son verdaderas o falsas. Justifique adecuadamente cada respuesta.

i) Si $L_a \cap L_b$ es regular no vacía y ambos definidos sobre el mismo alfabeto, entonces L_a es regular o L_b es regular

ii) Si $L_c \cap L_d$ es regular y L_c es regular, entonces L_d es regular

iii) Si L_e es recursivamente enumerable, entonces es infinito

iv) Sabiendo que la unión de dos lenguajes regulares es regular, es posible afirmar que la unión infinita de lenguajes regulares es también regular

v) La aplicación del algoritmo que convierte un AFND-ε → AFND, siempre mantiene el conjunto de estados finales

vi) La aplicación del algoritmo que convierte un AFND-ε → AFND, siempre mantiene el conjunto de estados

Solución

i) **Falso**. Sean $L_a=\{x:x=0^k 1^k, k\geq 0\}$ y $L_b=\{x:x=1^k 0^k, k\geq 0\}$
$L_a \cap L_b=\{\varepsilon\} \Rightarrow L_a \cap L_b$ es regular. Sin embargo, ni L_a ni L_b lo son.

ii) **Falso**. Sean $L_c=\{x:x=0^k, k\geq 0\}$ y $L_d=\{x:x=1^k 0^k, k\geq 0\}$ donde L_c es regular.
$L_c \cap L_d=\{\varepsilon\} \Rightarrow L_c \cap L_d$ es regular. L_d no es regular.

iii) **Falso**. $L_e=\{a\}$ es un lenguaje finito y, por la Jerarquía de Chomsky, es también Recursivamente enumerable. Por tanto, la implicancia no se cumple.

iv) **Falso**. Sea L_i una familia de lenguajes tal que $L_i=\{a^i b^i\}$. Es decir, que:

- $L_0=\{a^0 b^0\}=\{\varepsilon\}$
- $L_1=\{a^1 b^1\}=\{ab\}$
- $L_2=\{a^2 b^2\}=\{aabb\}$
- ...

Entonces, es sencillo ver que $U_{i=0}^{\infty} L_i=\{a^k b^k : k\in\mathbb{N}\}$ es el lenguaje Libre de Contexto no-regular visto en el teórico. Entonces no es posible afirmar ese enunciado.

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.**

v) **Falso**. Recordemos la definición del conjunto F' para este algoritmo:

$$F' = F \cup \{q0\} \; sii \; F \cap \varepsilon - cl(q_0) \neq \emptyset$$

Por tanto, en el caso de que el estado inicial sea final o que su épsilon clausura no contenga un estado final, el conjunto de estados finales se mantiene. Sin embargo todo AFND-ε cuyo estado inicial no cumpla eso, tendrá un AFND asociado donde $q_0 \in F'$. En ese caso, los conjuntos F y F' son diferentes.

vi) **Verdadero**. El pasaje AFND-ε → AFND solo modifica la función de transición y, posiblemente, el conjunto de estados finales por lo desarrollado en la parte anterior; sin embargo, no modifica el conjunto de estados Q.

b) Construya un autómata con salida de Mealy M: (Q,Σ,Λ,δ,λ,q_0), Σ={a,b}, Λ={X,Y},

λ : Q x (Σ ∪ {ε}) → (Λ ∪ {ε}) donde:

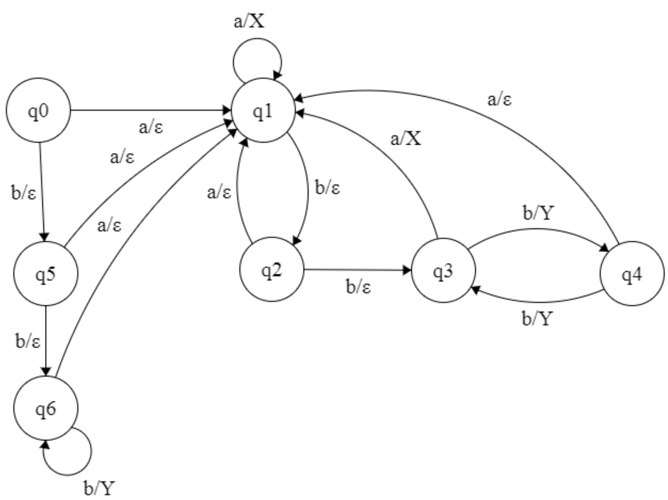
- imprime una X por cada secuencia de la forma $ab^{2n}a$, n≥0
- imprime una Y por cada secuencia de la forma bbb

Ejemplos:

Entrada	Salida
ε	ε
abbba	Y
aabbaaa	XXXX
aabbbbab	XYYX
ababb	ε

<u>Observación:</u> Tanto las b's como las a's pueden pertenecer a secuencias contiguas

Solución



En el circuito q_0-q_5-q_6 se contempla el caso de lectura de b's sin a's como prefijo.
A partir de q_1 se modela la situación donde el último símbolo leído fue una a.

- q_1 : el último símbolo leído fue una a
- q_2 : se lee la primera b en un bloque iniciado por una a
- q_3 : se lee la segunda b en un bloque iniciado por una a. **La cantidad de b's en el bloque es par**.

- q_4 : se lee la tercera b en un bloque iniciado por una a. A partir de ahora, cada b que venga estará enmarcada dentro de una sucesión de 3 b's, por lo que podrá imprimir una Y. **La cantidad de b's en el bloque es impar**.

Ejercicio 2 **[14 puntos]**

Sea L_2 = { w / w ∈ {0,1}* / w es de la forma $0^j 1^n 0^m$ / n < 2*j ; m impar ; n,m,j >0 } un lenguaje NO Regular.

a) Construya una gramática G_2 / L_2 = L(G_2). ¿Está simplificada? Justifique.

Solución

S → S 00 | A 0
A → 0A11 | 0A1 | 0A | 01

La generación de las tiras se hace en dos bloques:

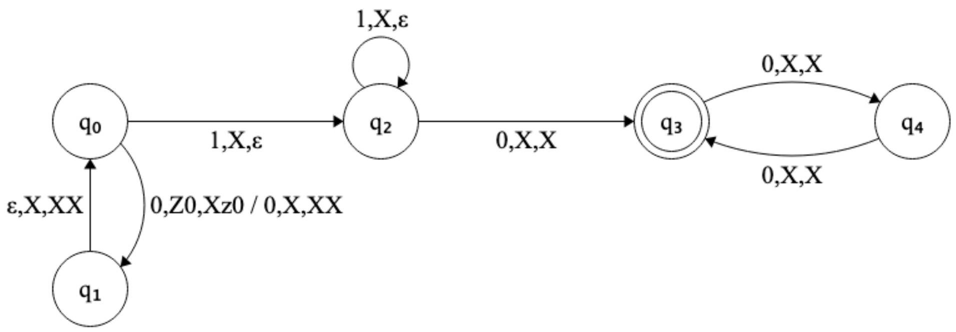
1. El correspondiente a la variable A, que controla la restricción n<2*j
2. El de la variable inicial S, que controla la imparidad del índice m

A su vez, la variable A permite:

- generar cantidad par de 1s
- generar cantidad impar de 1s
- generar una cantidad libre de 0s, dado que el incremento del índice j no rompe la restricción n<2*j
- terminar la producción de la tira de terminales asegurando que se quiebra la eventual igualdad n=2*j

La gramática **está simplificada** dado que todas sus variables son útiles y no hay producciones nulas ni unitarias.

b) Construya un autómata M_2 / L_2 = L(M_2). ¿Es determinista? Justifique.



El autómata **es determinista** dado que para cualquier estado, dado una entrada y un tope de stack, hay a lo sumo una única transición posible.

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.**

Ejercicio 3 **[18 puntos]**

Sea L_3 = { w / w ∈ {a,b,c,d}* / w es de la forma $a^{n+m} b^n c^m d^{n+m}$ / n >0, m ≥ 0 }

a) Clasifique L_3 según la Jerarquía de Chomsky. Justifique su respuesta.

El lenguaje L_3 es Recursivamente Enumerable no-Libre de Contexto. Haremos una prueba haciendo uso del contrarrecíproco del Pumping Lemma con el objetivo de probar que no es Libre de Contexto. En la siguiente parte se construirá una Máquina de Turing que compute el lenguaje, probando entonces, que es Recursivamente Enumerable.

Tomemos la tira $z=a^N b^N d^N$ que pertenece a L_3. Entonces las posibles familias de descomposiciones, son:

	a^N	b^N	d^N
i	vx		
ii	vx...	...x	
iii	v	x	
iv	v...	...vx	
v		vx	
vi		vx...	...x
vii		v	x
viii		v...	...vx
ix			vx

Familia i)

$u=a^p$
$v=a^q$ $q+r+s \leq N$
$w=a^r$ $q+s \geq 1$
$x=a^s$
$y=a^{N-p-q-r-s}b^N d^N$

$$z_i=a^{p+q.i+r+s.i+N-p-q-r-s}b^N d^N=a^{N+q(i-1)+s(i-1)}b^N d^N$$

por lo que tomando i=0: $z_0=a^{N-q-s}b^N d^N$

Como tenemos que q+s≥1 entonces N-(q+s)<N, por lo que no se cumple $|z_0|_a=|z_0|_d$. Por tanto,la tira z no pertenece a L_3.

El caso de la familia v) se prueba de igual forma, donde se reduce la cantidad de b's, provocando una violación a la cantidad con respecto a la esperada en las a's y d's.

Por último, el caso de la familia ix) también, donde la cantidad de d's es inferior a la de a's, rompiendo la misma condición que en este caso.

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.**

Familia ii)

$u=a^{N-p-q-r}$
$v=a^{p}$ $p+q+r+s \leq N$
$w=a^{q}$ $p+r+s\geq 1$
$x=a^{r}b^{s}$ $r\geq 1$ (para diferenciar del caso iii)
$y=b^{N-s}d^{N}$ $s\geq 1$ (para diferenciar del caso i)

$$z_i=a^{N-p-q-r+p.i+q}\left(a^r b^s\right)^i b^{N-s} d^N=a^{N+p(i-1)-r}\left(a^r b^s\right)^i b^{N-s} d^N$$

por lo que tomando i=2: $z_2=a^{N+p-r}a^r b^s a^r b^s b^{N-s} d^N$

Como tenemos que r≥1 y s≥1, entonces habrán a's y b's mezcladas, lo cual rompe la restricción de orden de los símbolos, por lo que la tira z no pertenece a L_3.

El caso de la familia vi) se prueba de igual forma, donde se mezclaran b's y d's.

Adicionalmente, los casos de las familias iv) y vi) se prueban de esta forma, mezclándose a's y b's o b's y d's, respectivamente.

Familia iii)

$u=a^{p}$
$v=a^{q}$ $N-p+r+s \leq N$
$w=a^{N-p-q}b^{r}$ $q+s\geq 1$
$x=b^{s}$
$y=b^{N-r-s}d^{N}$

$$z_i=a^{p+q.i+N-p-q}b^{r+s.i+N-r-s}d^N=a^{N+q(i-1)}b^{N+s(i-1)}d^N$$

por lo que tomando i=0: $z_0=a^{N-q}b^{N-s}d^N$

Al igual que en la familia i), como tenemos que q+s≥1, se cumple que al menos uno de ellos es mayor o igual a 1.

En el caso de que q≥1, la cantidad de a's no coincide con la cantidad de d's.

$$|z_i|_a=|z_i|_d \Leftrightarrow N-q=N$$
$$|z_i|_a=|z_i|_d=|z_i|_b+|z_i|_c \Leftrightarrow N=N-s+0 \Leftrightarrow N=N-s$$

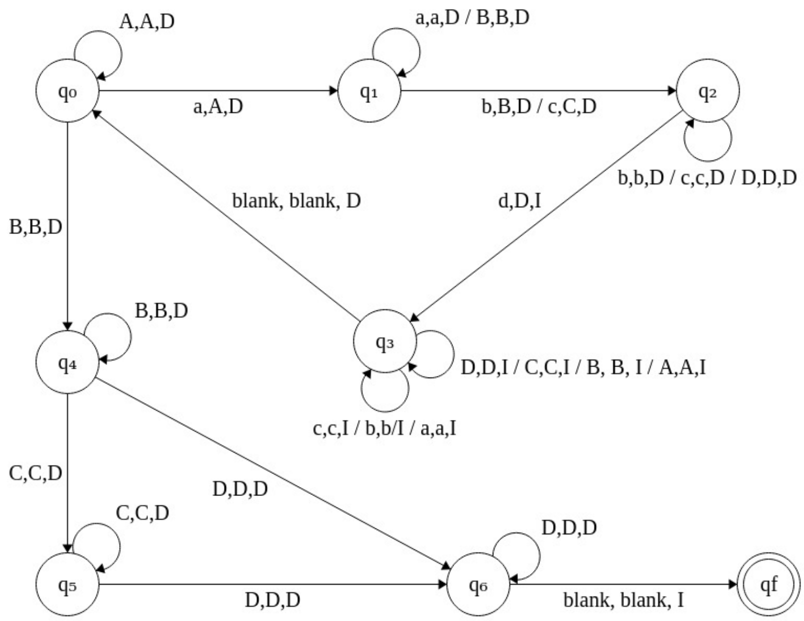
En el caso de que s≥1, la cantidad de b's no coincide con la cantidad esperadas de a's y d's.

Por lo cual, la tira z no pertenece a L_3 independientemente del valor de q y s.

El caso de la familia vii) se prueba de igual forma, donde las cantidades afectadas son de b's y d's (en lugar de a's y b's).

Se probó para una tira apropiada y todas sus descomposiciones posibles, que no pertenece al lenguaje, por tanto usando el contrarrecíproco del pumping lemma afirmamos que L_3 no es libre de contexto.

b) Construya un autómata M_3 / $L_3=L(M_3)$.



Se presenta entonces una Máquina de Turing para el lenguaje L_3. A continuación, una breve explicación su funcionamiento:

- los estados q_0, q_1 y q_2 se encargan de controlar la restricción dependencia en cantidad de los índices ($a^{n+m}\ b^n\ c^m\ d^{n+m}$)
- q_3 rebobina la cinta hasta el posicionarse en el primer símbolo escrito en ella.
- q_4 adelanta todas las marcas **B** comenzando con la etapa final de procesamiento, donde se chequea la correctitud de la tira en torno al orden relativo de los símbolos y la cantidad mínima de cada uno de ellos. Recordemos que a este estado se llega únicamente desde q_0 por lo que la presencia de la marca **A** obligatoria está asegurada.
- q_5 y q_6 procesan las marcas **C** y **D**, respectivamente. Cabe destacar que la opción de que el índice m valga 0 está contemplada con la transición directa de q_4 a q_6.