Teoría de Lenguajes

Soluciones

Ejercicio 1

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

a) El lenguaje L_{1a} = { x / x es de la forma a^p b^j c^m con $m \geq 0$, $j>0$, $m=2^p$} es libre de contexto.

Falso. El lenguaje L_{1a} No es Libre de Contexto y se probará usando el contra-recíproco del Pumping Lema para Lenguajes Libres de Contexto.

Dado N, elegimos $z = a^N bc^{2^\wedge N} \in L_{1a}$, tal que $|z| = N + 1 + 2^N \geq N$
Estudiamos todas las descomposiciones de z = uvwxy / $|vwx| \leq N$ y $|vx| \geq 1$

Familia	**a^N**	**b**	**$c^{2^\wedge N}$**
1	vx		
2	vx	x	x
3	v	v	vx
4	v		x
5			vx

Familia 1
$u = a^p$
$v = a^q$ $q+s \geq 1$
$w = a^r$ $q+r+s \leq N$
$x = a^s$
$y = a^{N-p-q-r-s}bc^{2^\wedge N}$

$z_i = a^p (a^q)^i a^r (a^s)^i a^{N-p-q-r-s}bc^{2^\wedge N} = a^{N(q+s)(i-1)}bc^{2^\wedge N}$
Tomando i=0 $z_0=a^{N-(q+s)}bc^{2^\wedge N}$, como $q+s \geq 1$ 2^cant_a(z_0) < cant_c(z_0), por lo tanto $z_0 \notin L_{1a}$

Para la **Familia 5** el razonamiento es similar, sólo que al elegir i=0 quedaría que 2^cant_a(z_0) > cant_c(z_0) con lo que $z_0 \notin L_{1a}$

Familia 2
$u = a^{N-p-q-r}$
$v = a^p$ $p+r+t+1 \geq 1$
$w = a^q$ $p+q+r+t+1 \leq N$
$x = a^r bc^t$
$y = c^{2^\wedge N - t}$

$z_i = a^{2^\wedge N-p-q-r} (a^p)^i a^q (a^r bc^t)^i c^{N-t}$
Tomando i=0 el problema es que z_0 no contiene al menos una **b**, por lo tanto $z_0 \notin L_2$
<u>Nota:</u> este caso exige que el substring x contenga siempre la "**b**", pero p,r,t podrían ser =0 todos o alguno de ellos

Familia 3
El estudio de ésta Familia es análogo al de la Familia 2 y con la misma justificación.

Familia 4
$u = a^{N-p-q}$
$v = a^p$ $p+s \geq 1$ (asumimos s>0, sino estamos en la Familia 2)
$w = a^q bc^t$ $p+q+1+t+s \leq N$
$x = c^s$
$y = c^{2^N - t - s}$

$z_i = a^{N-p-q} (a^p)^i a^q bc^t (c^s)^i c^{2^N-t-s} = a^{N+(i-1)p} bc^{2^N+(i-1)s}$

Tomando nuevamente i=2 $z_2 = a^{N-p} bc^{2^N+s}$

Se intentará demostrar que $2^N + s$ (que es la cantidad de c's) NO es potencia de 2

Se cumple que (la cantidad de c's) $2^N < 2^N + s$ porque s>0

Asimismo, $2^N + s \leq 2^N + N < 2^N + 2^N < 2 * 2^N = 2^{N+1}$ (porque $s < N < 2^N, \forall N$)

Por tanto $\nexists$ c donde la potencia de la cantidad de a's sea igual que la cantidad de c's y por lo tanto $z_0 \notin L_{1a}$

Como estas son todas la descomposiciones posibles que cumplen $|uwx| \leq N$ y $|vx| \geq 1$, se concluye que L_{1a} NO es un Lenguaje Libre de Contexto.

b) Sea L_{1b} un lenguaje Libre de Contexto no Regular. Entonces L_{1b}^r (reverso de L_{1b}) es Libre de Contexto no Regular.

Verdadero. Supongo que L_{1b}^r es Regular. Como los lenguajes regulares son cerrados bajo reverso, entonces L_{1b} sería regular. Pero L_{1b} NO es Regular, entonces L_{1b}^r NO es Regular. Es Libre de Contexto porque los Libres de Contexto son cerrados bajo reverso. Entonces L_{1b}^r es Libre de Contexto NO Regular.

c) Dado el lenguaje L_c = L(ab*a) entonces se cumple que **a R_{Lc} ba** siendo R_{Lc} la relación sobre lenguajes definida en el curso

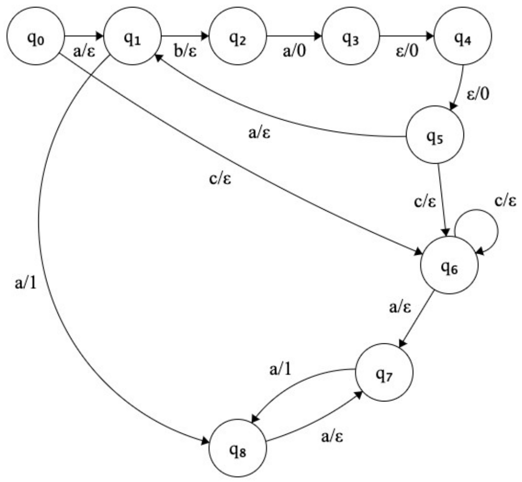
Falso. Si bien son dos tiras que no pertenecen al lenguaje L(ab*a), esas tiras están en la misma clase de equivalencia si $\forall z \in \Sigma^*$ $az \in L_c \wedge baz \in L_c$ o $az \notin L_c \wedge baz \notin L_c$
Si se toma z = **a** se tiene que **aa** $\in L_c \wedge$ **baa** $\notin L_c$

d) Todo lenguaje libre de contexto pero no regular puede ser generado por una gramática irrestricta.

Verdadero. Según la Jerarquía de Chomsky, los lenguajes Libres de Contexto (sean o no Regulares) son también lenguajes Recursivamente Enumerables y por lo tanto pueden ser generados por gramáticas irrestrictas.

Ejercicio 2

a) Construya un autómata con salida M_{2a}: $(Q, \Sigma, \Delta, \delta, \lambda, q_0)$ tal que dada una entrada de la forma $(aba)^p c^m a^{2k}$ produzca una salida de la forma $0^{3p}1^k$, con $p,m \geq 0$, $k>0$.
Considere $\Sigma=\{a,b,c\}$; $\Delta= \{0,1\}$; $\lambda: Q \times (\Sigma \cup \{\varepsilon\}) \rightarrow (\Delta \cup \{\varepsilon\})$

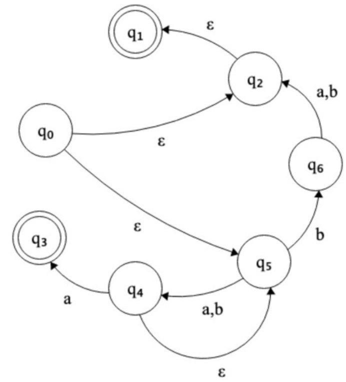


b) Dado el siguiente autómata finito M_{2b}:$(\{q0, q1, q2, q3, q4,q5,q6\}, \{a,b\}, \delta, q0, \{q1,q3\})$ siendo δ dado por:

	a	b	ε
q0	{}	{}	$\{q_2,q_5\}$
q1	{}	{}	{}
q2	{}	{}	$\{q_1\}$
q3	{}	{}	{}
q4	$\{q_3\}$	{}	$\{q_5\}$
q5	$\{q_4\}$	$\{q_4,q_6\}$	{}
q6	$\{q_2\}$	$\{q_2\}$	{}

i) Construya el autómata finito mínimo M_{2b}' / $L(M_{2b})=L(M_{2b}')$

El autómata M_{2b} es el siguiente:



Para el pasaje de AFND-ε → AFND, se aplica $\delta'(q,a) = \varepsilon\text{-clausura}(\tilde{\delta}(\varepsilon\text{-clausura}(q),a))$
Primero, calculamos todos los ε-clausura:
$\varepsilon\text{-cl}(q_0) = \{q_0, q_1, q_2, q_5\}$
$\varepsilon\text{-cl}(q_1) = \{q_1\}$
$\varepsilon\text{-cl}(q_2) = \{q_1, q_2\}$
$\varepsilon\text{-cl}(q_3) = \{q_3\}$
$\varepsilon\text{-cl}(q_4) = \{q_4, q_5\}$
$\varepsilon\text{-cl}(q_5) = \{q_5\}$
$\varepsilon\text{-cl}(q_6) = \{q_6\}$

y luego calculamos δ′:
$\delta'(q_0, a) = \varepsilon\text{-clausura}(\tilde{\delta}(\varepsilon\text{-clausura}(q_0),a)) = \varepsilon\text{-clausura}(\tilde{\delta}(\{q_0,q_1,q_2,q_5\},a)) = \varepsilon\text{-clausura}(\{q_4\}) = \{q_4,q_5\}$
$\delta'(q_0, b) = \varepsilon\text{-clausura}(\tilde{\delta}(\varepsilon\text{-clausura}(q_0),b)) = \varepsilon\text{-clausura}(\tilde{\delta}(\{q_0,q_1,q_2,q_5\},b)) = \varepsilon\text{-clausura}(\{q_4,q_6\}) = \{q_4,q_5,q_6\}$
$\delta'(q_1, a) = \varepsilon\text{-clausura}(\tilde{\delta}(\varepsilon\text{-clausura}(q_1),a)) = \varepsilon\text{-clausura}(\tilde{\delta}(\{q_1\},a)) = \varepsilon\text{-clausura}(\{\}) = \{\}$
$\delta'(q_1, b) = \varepsilon\text{-clausura}(\tilde{\delta}(\varepsilon\text{-clausura}(q_1),b)) = \varepsilon\text{-clausura}(\tilde{\delta}(\{q_1\},b)) = \varepsilon\text{-clausura}(\{\}) = \{\}$
$\delta'(q_2, a) = \varepsilon\text{-clausura}(\tilde{\delta}(\varepsilon\text{-clausura}(q_2),a)) = \varepsilon\text{-clausura}(\tilde{\delta}(\{q_1,q_2\},a)) = \varepsilon\text{-clausura}(\{\}) = \{\}$
$\delta'(q_2, b) = \varepsilon\text{-clausura}(\tilde{\delta}(\varepsilon\text{-clausura}(q_2),b)) = \varepsilon\text{-clausura}(\tilde{\delta}(\{q_1,q_2\},b)) = \varepsilon\text{-clausura}(\{\}) = \{\}$
$\delta'(q_3, a) = \varepsilon\text{-clausura}(\tilde{\delta}(\varepsilon\text{-clausura}(q_3),a)) = \varepsilon\text{-clausura}(\tilde{\delta}(\{q_3\},a)) = \varepsilon\text{-clausura}(\{\}) = \{\}$
$\delta'(q_3, b) = \varepsilon\text{-clausura}(\tilde{\delta}(\varepsilon\text{-clausura}(q_3),b)) = \varepsilon\text{-clausura}(\tilde{\delta}(\{q_3\},b)) = \varepsilon\text{-clausura}(\{\}) = \{\}$
$\delta'(q_4, a) = \varepsilon\text{-clausura}(\tilde{\delta}(\varepsilon\text{-clausura}(q_4),a)) = \varepsilon\text{-clausura}(\tilde{\delta}(\{q_4,q_5\},a)) = \varepsilon\text{-clausura}(\{q_3,q_4\}) = \{q_3,q_4,q_5\}$
$\delta'(q_4, b) = \varepsilon\text{-clausura}(\tilde{\delta}(\varepsilon\text{-clausura}(q_4),b)) = \varepsilon\text{-clausura}(\tilde{\delta}(\{q_4,q_5\},b)) = \varepsilon\text{-clausura}(\{q_4,q_6\}) = \{q_4,q_5,q_6\}$
$\delta'(q_5, a) = \varepsilon\text{-clausura}(\tilde{\delta}(\varepsilon\text{-clausura}(q_5),a)) = \varepsilon\text{-clausura}(\tilde{\delta}(\{q_5\},a)) = \varepsilon\text{-clausura}(\{q_4\}) = \{q_4,q_5\}$
$\delta'(q_5, b) = \varepsilon\text{-clausura}(\tilde{\delta}(\varepsilon\text{-clausura}(q_5),b)) = \varepsilon\text{-clausura}(\tilde{\delta}(\{q_5\},b)) = \varepsilon\text{-clausura}(\{q_4,q_6\}) = \{q_4,q_5,q_6\}$
$\delta'(q_6, a) = \varepsilon\text{-clausura}(\tilde{\delta}(\varepsilon\text{-clausura}(q_6),a)) = \varepsilon\text{-clausura}(\tilde{\delta}(\{q_6\},a)) = \varepsilon\text{-clausura}(\{q_2\}) = \{q_1,q_2\}$
$\delta'(q_6, b) = \varepsilon\text{-clausura}(\tilde{\delta}(\varepsilon\text{-clausura}(q_6),b)) = \varepsilon\text{-clausura}(\tilde{\delta}(\{q_6\},b)) = \varepsilon\text{-clausura}(\{q_2\}) = \{q_1,q_2\}$

Entonces, tenemos:

δ′	**a**	**b**
q0	$\{q_4,q_5\}$	$\{q_4,q_5,q_6\}$
q1	{}	{}
q2	{}	{}
q3	{}	{}
q4	$\{q_3,q_4,q_5\}$	$\{q_4,q_5,q_6\}$
q5	$\{q_4,q_5\}$	$\{q_4,q_5,q_6\}$
q6	$\{q_1,q_2\}$	$\{q_1,q_2\}$

con $q_0'=q_0$ y $F'=\{q_0,q_1,q_3\}$

Ahora hacemos el pasaje AFND → AFD

δ''	**a**	**b**
[q_0]	[q_4q_5]	[$q_4q_5q_6$]
[q_4q_5]	[$q_3q_4q_5$]	[$q_4q_5q_6$]
[$q_4q_5q_6$]	[$q_1q_2q_3q_4q_5$]	[$q_1q_2q_4q_5q_6$]
[$q_3q_4q_5$]	[$q_3q_4q_5$]	[$q_4q_5q_6$]
[$q_1q_2q_3q_4q_5$]	[$q_3q_4q_5$]	[$q_4q_5q_6$]
[$q_1q_2q_4q_5q_6$]	[$q_1q_2q_3q_4q_5$]	[$q_1q_2q_4q_5q_6$]

Renombrando los estados tenemos:

δ''	**a**	**b**
q_0	q_1	q_2
q_1	q_3	q_2
q_2	q_4	q_5
q_3	q_3	q_2
q_4	q_3	q_2
q_5	q_4	q_5

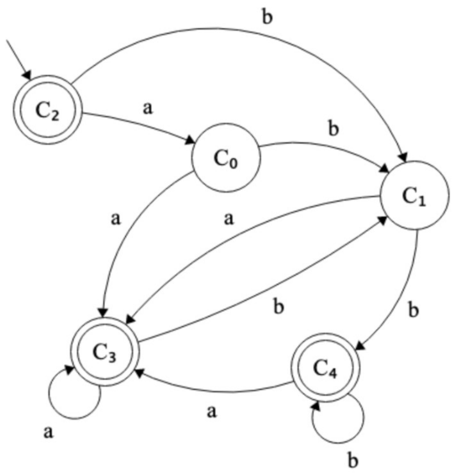
Con $q_0''=q_0$ y $F''=\{q_0,q_3,q_4,q_5\}$

Como δ'' es completa, pasamos a minimizar:
$Q\text{-}F = \{q_1,q_2\}$, $F=\{q_0,q_3,q_4,q_5\}$

$\Pi_0 = \{q_1,q_2\}\ \{q_0,q_3,q_4,q_5\}$
$\Pi_1 = \{q_1\}\ \{q_2\}\ \{q_0\}\ \{q_3,q_4\}\ \{q_5\}$
$\Pi_2 = \{q_1\}\ \{q_2\}\ \{q_0\}\ \{q_3,q_4\}\ \{q_5\}$
Renombramos los estados:
$\{q_1\} \to C_0$
$\{q_2\} \to C_1$
$\{q_0\} \to C_2$
$\{q_3,q_4\} \to C_3$
$\{q_5\} \to C_4$
entonces tenemos:

δ'''	**a**	**b**
C_2	C_0	C_1
C_0	C_3	C_1
C_1	C_3	C_4
C_3	C_3	C_1
C_4	C_3	C_4

Con $q_0'''=C_2$ y $F'''=\{C_2,C_3,C_4\}$



ii) Construya una Gramática G_{2b} / $L(G_{2b}) = L(M_{2b})$

Como tenemos un autómata finito determinista, podemos obtener fácilmente una gramática regular.

- Por cada estado del autómata tendremos un símbolo variable en la gramática. Al estado inicial lo representa el símbolo inicial.
- Por cada símbolo del alfabeto se tiene un símbolo terminal en la gramática.
- Cada transición del autómata será una producción.
- Aquellas variables que representen estados finales se les agrega una producción epsilon.

Entonces, la gramática G_{2b} pedida es: $G_{2b} = (V,T,P,S)$

$V=\{S,W,X,Y,Z\}$ (asociamos $S \to C_2$, $W \to C_0$, $X \to C_1$, $Y \to C_3$, $Z \to C_4$)
$T=\{a,b\}$
$P = \{ S \to aW \mid bX \mid \varepsilon$
$W \to aY \mid bX$
$X \to aY \mid bZ$
$Y \to aY \mid bX \mid \varepsilon$
$Z \to aY \mid bZ \mid \varepsilon \}$

Observar que la gramática planteada es una gramática lineal derecha.

Ejercicio 3

Sea $L_3 = \{ x \mid x$ es de la forma $a^{3n}b^k (bab^k)^n : k\geq 1 \wedge n \in \{0,1\} \}$ un lenguaje no regular

a) Si fuera posible, construya una gramática simplificada G_3 / $L(G_3)=L_3$.
b) Construya un autómata M_3 / $L(M_3) = L_3$. ¿Es determinista? Justifique.
c) Sugiera algún cambio en la definición de L_3 respecto a las variables k y/o n para que el lenguaje sea libre de contexto. Justifique.
d) ¿Qué cambio le haría a L_3 (respecto a las variables k y/o n) para que sea regular? Justifique.

a) Por la letra se sabe que no es un lenguaje regular, y se probará que es libre de contexto al poder generarse con la siguiente gramática libre de contexto.

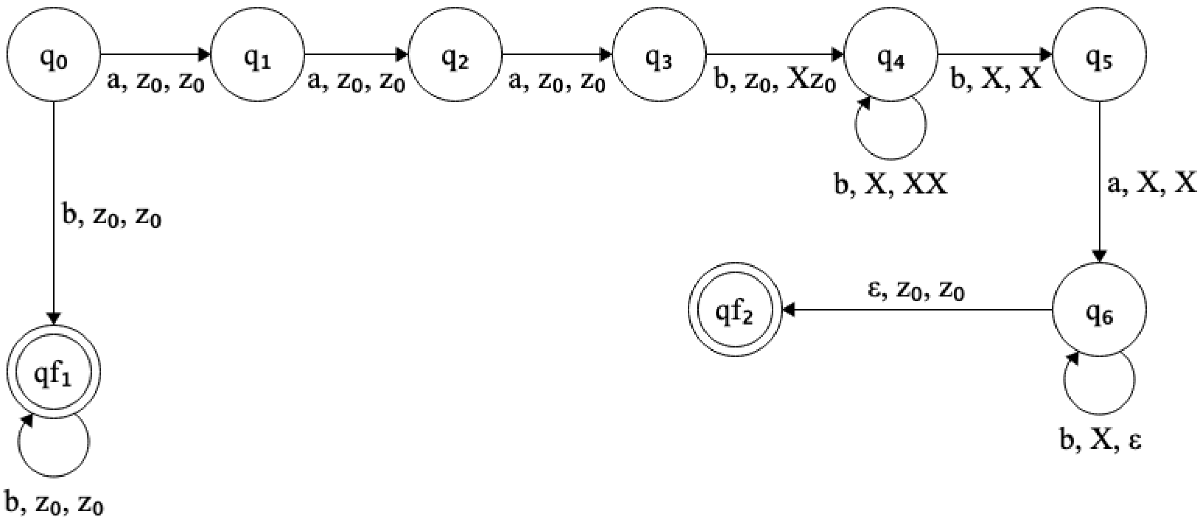
$S \to X \mid aaaY$
$X \to bX \mid b$
$Y \to bbab \mid bYb$

Como la gramática dada NO está simplificada, se aplica el algoritmo de simplificación visto en el curso quedando finalmente:

$S \to bX \mid b \mid aaaY$
$X \to bX \mid b$
$Y \to bbab \mid bYb$

donde no hay producciones epsilon (no las tenía la original), no hay unitarias (se eliminó la que había $S \to X$ sustituyéndola por $S \to bX \mid b$) y todas las variables son últiles.

b) Se construye un autómata push-down que acepta por estado final. El autómata no es determinista, ya que en el estado q_4 hay dos transiciones posibles para la entrada "b" y el tope de pila "X": una hacia q_4 y otra hacia q_5.



c) No hay que hacer ningún cambio. El lenguaje ya es libre de contexto y fue probado al dar una GLC y un APD para él.

d) Se podría fijar por ejemplo $n=0$, con lo que se tendría el lenguaje $L' = \{b^k: k \geq 1\}$ que es regular, ya que $L' = L(bb^*)$.

Ejercicio 4

Sea el alfabeto $\Sigma = \{a,b\}$.
Construya una Máquina de Turing que compute una función que recibe una tira de la forma x#n donde $x \in \Sigma^*$ y $n > 0$ es un natural en unario, y devuelve x^n, donde x^n denota a la tira x repetida n veces.

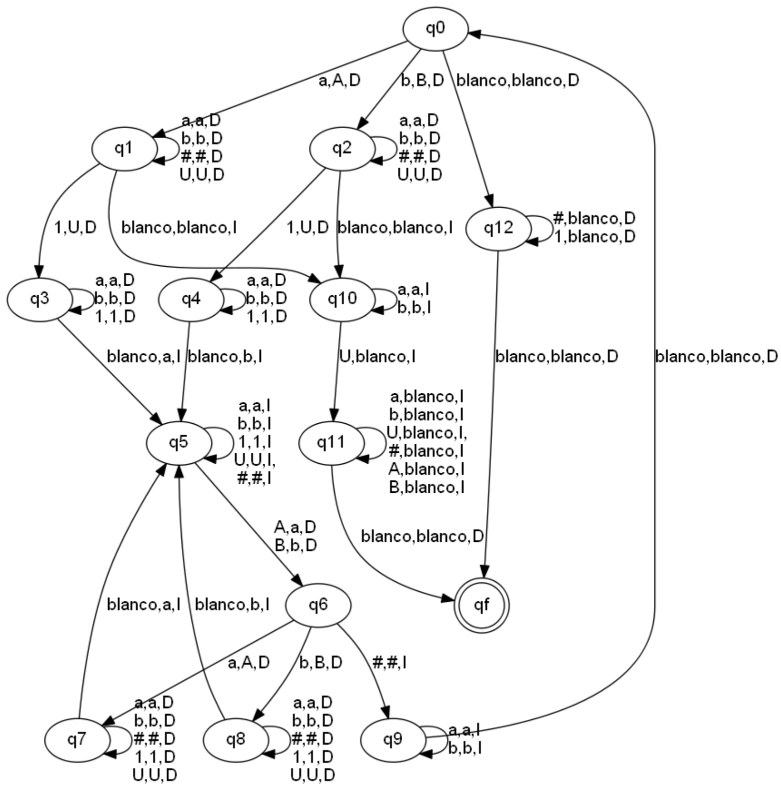
Ejemplos:

Entrada	Salida
ε#1111	ε
abba#1	abba
bb#11	bbbb
baa#111	baabaabaa

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.**

Nota: en la configuración final de la MT, en la cinta sólo debe quedar el resultado.

La idea de la MT es replicar la tira x tantas veces como 1's figuren en n.



Ejercicio 4 - Teoría de la Programación I

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

a) El problema de satisfacer una fórmula booleana [SAT] es un problema NP-Completo.

Verdadero. Es el enunciado del Teorema de Satisfacibilidad booleana de Cook

b) Si A es un problema NP y $A \propto SAT \Rightarrow$ A es NP-completo.

Falso. Todo problema NP se transforma en tiempo polinomial a SAT (es justamente el Teorema de Cook). Sin embargo, no se puede afirmar nada, ya que si $P \neq NP$, habría problemas en P que no son NP-completos.

c) La función $h(i,j,k) = \begin{cases} 1 & \text{si } \langle I_x(i),k\rangle \downarrow \text{ en j pasos } \vee \langle I_x(j),k\rangle \downarrow \text{ en i pasos} \\ 0 & \text{en caso contrario} \end{cases}$

NO es computable.

Falso. A continuación se escribe un programa en P que la computa:

```
PROGRAM(<i,j,k>)
    X0:=0;
    X1:=EVAL_PROG_STEP(i,k,j);
    X2:=EVAL_PROG_STEP(j,k,i);

    IF (FST(X1)=1 OR FST(X2)=1) THEN
        X0:=1
    FI
RESULT(X0)  .
```

$\neq$

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.**