Solucions a l'examen final del 07/01/2010

1. (2,5 punts) Doneu una gramàtica sobre l'alfabet $\{a, b, c\}$ que generi els mots de la forma xcy on $x, y \in \{a, b\}^*$, tals que x no sigui prefix de y. Cal que la justificació del resultat inclogui l'explicitació de quin és el llenguatge generat per cada una de les variables que constitueixen la gramàtica proposada. Es considerarà com a mèrit addicional que la gramàtica sigui lineal (és a dir, que els costats drets de cada producció continguin com a molt una variable).
Ajuda: Considereu la relació entre el llenguatge plantejat i el definit formalment per l'expressió següent:

$$\{xcy \mid \exists x_1, x_2, y_1, y_2 \in \{a, b\}^* \ \exists s, t \in \{a, b\} \ (x = x_1 s x_2 \wedge y = y_1 t y_2 \wedge |x_1| = |y_1| \wedge s \neq t)\}.$$

Resposta: Un mot x no és prefix d'un mot y si es dóna una de les dues condicions següents, que no són excloents l'una de l'altra: o bé x és més llarg que y, o bé hi ha almenys un símbol de x que és diferent del símbol de y que figura a la mateixa posició. El conjunt de mots que satisfan la segona condició és precisament el llenguatge que apareix formulat a l'ajuda. Descompondrem el llenguatge donat en reunió d'aquests dos llenguatges fent

$$S \to A \mid B$$

on A ha de generar el llenguatge $\{xcy \mid |x| > |y|\}$ i B ha de generar el llenguatge donat a l'ajuda, que podem expressar alternativament així:

$$\bigcup_{n \geq 0} (a+b)^n a (a+b)^* c (a+b)^n b (a+b)^* \ \cup \ \bigcup_{n \geq 0} (a+b)^n b (a+b)^* c (a+b)^n a (a+b)^*.$$

La variable A genera $\{xcy \mid |x| > |y|\}$ així:

$$A \to aAa \mid aAb \mid bAa \mid bAb \mid aA \mid bA \mid ac \mid bc,$$

on les quatre primeres produccions generen el submot y i una part de x d'igual longitud que y, mentre que les quatre últimes s'encarreguen de generar el sufix de x que rebassa la longitud de y.

Per altra banda, podem expressar el llenguatge que li toca generar a la variable B així: $(L_a b \cup L_b a)(a+b)^*$ on hem fet $L_a = \bigcup_{n \geq 0} (a+b)^n a (a+b)^* c (a+b)^n$ i $L_b = \bigcup_{n \geq 0} (a+b)^n b (a+b)^* c (a+b)^n$.

Si definim una variable C encarregada de generar L_a i una variable D encarregada de generar L_b, les produccions de B són:

$$B \to Ba \mid Bb \mid Cb \mid Da.$$

És a dir, que la variable B comença generant per la dreta $(a+b)^*$ amb les dues primeres produccions i després genera o bé $L_a b$ amb la tercera, o bé $L_b a$ amb la quarta. Ara, la manera com C i D generen els seus respectius llenguatges és simètrica l'una de l'altra. Ens ajudem d'una última variable E que s'encarrega de genrar la part $(a+b)^*$ que apareix al centre dels dos llenguatges. Les produccions són:

$$C \to aCa \mid aCb \mid bCa \mid bCb \mid aEc$$

$$D \to aDa \mid aDb \mid bDa \mid bDb \mid bEc$$

$$E \to aE \mid bE \mid \lambda.$$

2. (3 punts) Trobeu el mínim DFA que reconeix el llenguatge format pels mots sobre l'alfabet $\{a, b\}$ tals que després de qualsevol ocurrència del submot bb, el nombre de a's que queden fins al final del mot és un nombre parell. Justifiqueu el resultat.

Resposta (simplificada):

Formalitzem aquest llenguatge com

$$L = \{w \in \{a,b\}^* \mid \forall x, y : (w = xbby \Rightarrow |y|_a \in \dot{2})\}$$

El seu complementari queda expressat com

$$\bar{L} = \{w \in \{a,b\}^* \mid \exists x, y : (w = xbby \wedge |y|_a \notin \dot{2})\}$$

Aquest llenguatge es pot expressar també com la següent concatenació:

$$\bar{L} = \{a,b\}^* bb \{y \big| \ |y|_a \notin \dot{2}\}$$

Per a cada subllenguatge és fàcil obtenir el corresponent DFA. Si els concatenem de la manera habitual i fem l'eliminació de λ-produccions obtenim el seguent NFA:

	a	b
$\to q_0$	q_0	q_0q_1
q_1	$\emptyset$	q_2
q_2	q_3	q_2
$\dagger q_3$	q_2	q_3

Si determinitzem, minimitzem i complementem, obtenim finalment:

	a	b
$\to \dagger q_0$	q_0	q_1
$\dagger q_1$	q_0	q_2
$\dagger q_2$	q_3	q_2
q_3	q_2	q_4
q_4	q_2	q_5
q_5	q_5	q_5

Resposta alternativa (simplificada):

El llenguatge revessat de L és:

$$L^R = \{w \in \{a,b\}^* \mid \forall x, y : (w = xbby \Rightarrow |x|_a \in \dot{2})\}$$

Per a aquest llenguatge resulta molt senzill obtenir directament un DFA:

	a	b
$\rightarrow \dagger q_0$	q_1	q_0
$\dagger q_1$	q_0	q_2
$\dagger q_2$	q_0	q_3
q_3	q_3	q_3

Aplicant les operacions de revessar, determinitzar i minimitzar, tornem a obtenir el DFA mínim solució.

3. (1 punt) Doneu dos llenguatges (ben diferents) de cada una de les classes següents:

a) no semidicidibles

b) semidecidibles però no decidibles

c) decidibles però no incontextuals

d) incontextuals però no regulars

e) regulars però no finits

Resposta:

a) $\{x \mid M_x(x) \uparrow\}$ $\quad$ $\{x \mid \forall y : M_x(y) \downarrow\}$

b) $\{x \mid M_x(x) \downarrow\}$ $\quad$ $\{\langle u, v, R\rangle \mid u \rightarrow_R^* v\}$

c) $\{a^n b^n c^n \mid n \geq 0\}$ $\quad$ $\{ww \mid w \in \{a,b\}^*\}$

d) $\{a^n b^n \mid n \geq 0\}$ $\quad$ $\{ww^R \mid w \in \{a,b\}^*\}$

e) $\{a,b\}^*$ $\quad$ $\{w \in \{0,1\}^* \mid \texttt{valor}_2(w) \in \dot{3}\}$

4. (1,5 punts) Classifiqueu com a decidible, semi-decidible però no decidible, o no semi-decidible el problema de determinar si un programa donat implementa la funció identitat. Formalment, es tracta de classificar el conjunt

$$\{x \mid \forall z\ \varphi_x(z) \downarrow\ \wedge \varphi_x(z) = z\}.$$

Resposta: Sigui L el llenguatge de l'enunciat. Demostrem que L no és semi-decidible proposant la següent reducció des de $\bar{K}$. Per cada natural x, la nostra reducció R genera el natural que codifica el següent programa:

entrada y

  Si $M_x(x)$ s'atura en y passos llavors sortida 1

  Si no sortida y

La nostra reducció és clarament computable. Resta veure que la reducció preserva la pertinença a tots dos conjunts $\bar{K}$ i L. Si un cert x pertany a $\bar{K}$ llavors $M_x(x)$ no s'atura i el programa anterior dona sortida y per a cada entrada y, de manera que $\varphi_{R(x)}$ és la funció identitat, i per tant $R(x)$ pertany a L. En cas contrari, si un cert x no pertany a $\bar{K}$ llavors $M_x(x)$ s'atura en un cert nombre de passos t, i el programa anterior dona sortida 1 per a qualsevol y major o igual a t, de manera que $\varphi_{R(x)}$ no és la funció identitat, i per tant $R(x)$ no pertany a L. Tot plegat justifica que la reducció és correcta, i per tant que L no és semi-decidible.

5. (2 punts) Demostreu que el Problema de la Correspondència de Post continua essent indecidible encara que el considerem restringit a mots sobre l'alfabet $\{0, 1\}$. Demostreu, en canvi, que si el restringim a un alfabet uniliteral, aleshores és decidible.

Resposta:

1a part: Anomenen PCP_{01} a la restricció de PCP d'aquest problema. Reduïm PCP a PCP_{01} de la següent manera. Sigui $\langle u_1, v_1, \ldots, u_n, v_n\rangle$ una entrada de PCP. Sigui $\{a_1, \ldots, a_m\}$ l'alfabet utilitzat en aquesta entrada. Definim el següent morfisme $\sigma : \{a_1, \ldots, a_m\}^* \to \{0, 1\}^*$ símbol a símbol així: $\sigma(a_i) = 0^i 1$. Clarament, σ és un morfisme injectiu, i per tant una funció total i injectiva. La sortida de la nostra reducció és $\langle\sigma(u_1), \sigma(v_1), \ldots, \sigma(u_n), \sigma(v_n)\rangle$.

La reducció que hem definit és clarament computable. Queda per veure que preserva la resposta entre tots dos problemes. Sigui $\langle u_1, v_1, \ldots, u_n, v_n\rangle$ una entrada de PCP amb resposta afirmativa. Llavors, existeixen $1 \leq i_1, \ldots, i_k \leq n$ amb $k \geq 1$ tals que $u_{i_1} \cdots u_{i_k} = v_{i_1} \cdots v_{i_k}$ es compleix. Com que σ és una funció total, també es compleix $\sigma(u_{i_1} \cdots u_{i_k}) = \sigma(v_{i_1} \cdots v_{i_k})$. Com que σ és un morfisme, també es compleix $\sigma(u_{i_1}) \cdots \sigma(u_{i_k}) = \sigma(v_{i_1}) \cdots \sigma(v_{i_k})$, i aixo conclou que es preserva la resposta afirmativa d'esquerra a dreta.

Sigui ara $\langle u_1, v_1, \ldots, u_n, v_n\rangle$ una entrada de PCP tal que $\langle\sigma(u_1), \sigma(v_1), \ldots, \sigma(u_n), \sigma(v_n)\rangle$. té resposta afirmativa a PCP_{01}. Llavors, existeixen $1 \leq i_1, \ldots, i_k \leq n$ amb $k \geq 1$ tals que $\sigma(u_{i_1}) \cdots \sigma(u_{i_k}) = \sigma(v_{i_1}) \cdots \sigma(v_{i_k})$ es compleix. Com que σ és morfisme, també es compleix $\sigma(u_{i_1} \cdots u_{i_k}) = \sigma(v_{i_1} \cdots v_{i_k})$. Com que σ és injectiu, també es compleix $u_{i_1} \cdots u_{i_k} = v_{i_1} \cdots v_{i_k}$, i aixo conclou que es preserva la resposta afirmativa de dreta a esquerra.

2a part: Per decidir si un PCP uniliteral té o no té solució només cal veure si tots els mots components d'una de les llistes són més llargs que els seus corresponents de l'altra llista. Si és així, el problema no té solució; altrament, sí que en té.

És clar que si tots els mots d'una de les llistes són més llargs que els seus parells de l'altra, no hi pot haver solució. Altrament, o bé hi ha algun parell de mots d'igual longitud (és a dir, iguals) i la tria d'aquest parell és una solució, o bé hi ha dos parells (a^i, a^j) i (a^k, a^l) (suposant que a és el símbol únic de l'alfabet) tals que $i < j$ i $k > l$. En aquest cas, una solució consisteix a prendre $k - l$ vegades el primer parell seguit de $j - i$ vegades el segon.