PROBLEMES RESOLTS (DE NIVELL D'EXAMEN)

(Parcial d'Abril-2005)

Problema 1 Considerem una variant del WP, que designarem com a W1P, en què almenys un dels dos termes de cada regla de reescriptura (un dels components u_i o v_i de cada parell) estigui reduït a un únic símbol, és a dir, per a cada regla de reescriptura (u_i, v_i) se satisfà que $u_i \in \Sigma$ o $v_i \in \Sigma$ on Σ és l'alfabet de R. Demostreu que W1P és indecidible fent la reducció de WP a W1P.

Resposta: La següent transformació redueix WP a W1P.

$$\langle u_1 \to v_1, \ldots, u_n \to v_n, u, v\rangle \mapsto \langle u_1 \to \$_1, \$_1 \to v_1, \ldots, u_n \to \$_n, \$_n \to v_n, u, v\rangle$$

On el conjunt $\{\$_1, \ldots, \$_n\}$, que anomenarem $\Sigma_\$$, s'ha escollit de manera que contingui símbols nous no inclosos en l'alfabet Σ utilitzat per les regles de partida.

La transformació és computable, i està clar que qualsevol derivació amb el sistema original es pot simular amb el corresponent sistema de reescriptura imatge.

Respecte a la direcció contraria, definim primer una mica de notació, per facilitar les explicacions. Sigui R el sistema original, i $R_{\to\$} \cup R_{\$\to}$ el sistema imatge, on els noms denoten el que escau més intuitiu.

Utilitzant només regles de $R_{\$\to}$, qualsevol paraula $w \in (\Sigma \cup \Sigma_\$)^*$ pot ser reescrita en una paraula $w' \in \Sigma^*$, que està unívocament determinada, i que denotarem com a $R_{\$\to}(w)$.

Donat un pas de reescriptura qualsevol $s \to_{R_{\to\$} \cup R_{\$\to}} t$, o bé és amb una regla de $R_{\$\to}$, i llavors $R_{\$\to}(s) = R_{\$\to}(t)$, o bé és amb una regla de $R_{\to\$}$, i llavors $R_{\$\to}(s) \to_R R_{\$\to}(t)$. En qualsevol cas, tenim que $R_{\$\to}(s) \to_R^* R_{\$\to}(t)$.

Per tant, si suposem que existeix una derivació $u = s_1 \to_{R_{\to\$} \cup R_{\$\to}} \ldots \to_{R_{\to\$} \cup R_{\$\to}} s_k = v$, també n'existirá una de la forma $R_{\$\to}(u) = R_{\$\to}(s_1) \to_R^* \ldots \to_R^* R_{\$\to}(s_k) = R_{\$\to}(v)$, Però com que $u = R_{\$\to}(u)$, i $v = R_{\$\to}(v)$, l'anterior és una derivació de u a v amb R.

Problema 2 Considereu una segona variant del WP, que designarem com a W2P, en què l'entrada està formada per una llista finita R' de regles de reescriptura, dos mots u' i v' i un símbol a que no forma part de u' ni de v'. Es tracta de determinar si és possible passar de u' a $v'av'$ utilitzant les regles de R'. Feu, de nou, la reducció de WP a W2P.

Resposta: Reduïm WP a W2P amb la següent transformació.

$$\langle u_1 \to v_1, \ldots, u_n \to v_n, u, v\rangle \mapsto \langle u_1 \to v_1, \ldots, u_n \to v_n, v \to v\$v, u, v, \$\rangle$$

On \$ s'ha escollit com a un símbol nou.

Anomenem R al sistema original. Si $u \to_R^* v$, llavors està clar que $u \to_R^* v \to_{v \to v\$v} v\$v$.

Vice-versa, suposem que $u \to_{R \cup \{v \to v\$v\}}^* v\$v$. Per la forma de les regles, la regla $v \to v\$v$ s'ha d'haver utilitzat exactament una vegada, ja que \$ ha d'acabar apareixent, i no pot ser esborrat. Per tant, la derivació és de la forma $u \to_R^* u_1 v u_2 \to_{v \to v\$v} u_1 v\$v u_2 \to_R^* v\$v$. Donat que \$ no apareix en les regles de R, necessàriament existeixen dues subderivacions $u_1 v \to_R^* v$ i $v u_2 \to_R^* v$, i tot plegat implica $u \to_R^* u_1 v u_2 \to_R^* v u_2 \to_R^* v$.

Solució alternativa (simplificada):

La següent transformació també funciona.

$$\langle u_1 \to v_1, \ldots, u_n \to v_n, u, v\rangle \mapsto \langle u_1 \to v_1, \ldots, u_n \to v_n, \$v\$ \to \#\#'\#, \$u\$, \#, \#'\rangle$$

On $\#, \#', \$$ s'han escollit com a símbols nous que no apareixien en el sistema i paraules inicials.

Problema 3 Considerem el problema següent: donades dues TM M i N, i donat un enter $n > 0$, determinar si els dominis de les funcions computades per M i N contenen almenys n elements en comú. És a dir, si $\| \mathsf{Dom}\, \mathrm{f}_M \cap \mathsf{Dom}\, \mathrm{f}_N \| \geqslant n$.

1. Formalitzeu aquest problema com un llenguatge C.
2. Demostreu que C és enumerable recursivament.
3. Demostreu que C no és decidible.

Resposta: El conjunt queda definit com segueix.

$$C = \{\langle x, y, n\rangle \mid \| \mathsf{Dom}\, \phi_x \cap \mathsf{Dom}\, \phi_y \| \geqslant n\}$$

El següent procés semidecideix aquest conjunt.

```
entrada ⟨x,y,n⟩
t:=1
Executar indefinidament:
   comptador:=0
   Desde i:=0 fins t fer
      Simular Mx(i) durant t passos
      Simular My(i) durant t passos
      Si totes dues execucions aturen i accepten llavors comptador:=comptador+1
   Si comptador⩾ n llavors acceptar
   t:=t+1
```

Per a demostrar que no és decidible, es pot reduïr desde K de la següent manera, on hem suposat que c és un nombre que codifica una màquina qualsevol que s'atura i accepta amb totes les entrades.

$$x \mapsto \langle f(x), c, 1\rangle$$

On f és la funció que, a partir de x, dona com a resultat la codificació de la màquina següent.

```
entrada y
   Simular Mx(x)
   acceptar
```

Les màquines $f(x)$ i c acceptaran més o igual que $n = 1$ entrades en comú si i només si x s'atura amb entrada ell mateix.

Problema 4 Considereu el problema següent: donada una TM M, determinar si existeix alguna entrada de longitud parella amb la qual M s'atura.

1. Formalitzeu aquest problema com un llenguatge C.
2. Demostreu que C és enumerable recursivament.
3. Demostreu que C no és decidible.

Resposta: Donem la següent formalització.

$$C = \{x \mid \exists y : (|y| = \dot{2} \wedge M_x(y) \downarrow)\}$$

On $|y|$ s'ha d'interpretar com la mida de la codificació del natural y.

El següent procés semi-decideix aquest conjunt:

```
entrada ⟨x⟩
t:=1
Executar indefinidament:
   Desde i:=0 fins t fer
      Si |i| és parell llavors
         Simular Mx(i) durant t passos
         Si la execució s'atura llavors acceptar
   t:=t+1
```

Es pot reduïr $\mathcal{K}$ a C amb la transformació que, donat `x`, genera el programa següent.

```
entrada y
   Simular Mx(x)
   acceptar
```

`x`$\in \mathcal{K}$ si i només si `M`$_x$`(x)` s'atura si i només si el programa anterior s'atura amb una entrada de longitud parell.

Problema 5 Sigui

$$f(x) = \begin{cases} M_x(x) + 1 & \text{si } M_x(x)\downarrow \\ \text{indefinida,} & \text{altrament.} \end{cases}$$

1. Demostreu que f és una funció computable.
2. Sigui c la codificació de la TM que computa f. Demostreu que $M_c(c)\uparrow$.

Resposta: La funció f es pot computar com segueix.

```
entrada x
   Simular Mx(x) i si acaba, guardar el resutat en y
   sortida y+1
```

Suposem que $M_c(c) \downarrow$. Per la definició de f tenim que $f(c) = M_c(c) + 1$. Però com que c computa f resulta que $f(c) = M_c(c)$, de lo que resulta $M_c(c) = M_c(c) + 1$, i per tant $0 = 1$: contradicció.

(Examen de Juny-2005)

Problema 1 Demostreu que les tres variants següents del problema de la correspondència de Post també són indecidibles.

a) (PCP amb nombre senar de regles (PCP1))

Donades dues llistes de longitud n, on n és un nombre senar, $A = (x_1, \ldots, x_n)$ i $B = (y_1, \ldots, y_n)$, de mots sobre un cert alfabet Σ, determinar si hi ha una seqüència finita no buida d'enters compresos entre 1 i n, $(i_1, \ldots, i_r)$, amb $r \geqslant 1$ tal que $x_{i_1} \cdots x_{i_r} = y_{i_1} \cdots y_{i_r}$.

b) (PCP bipartit (PCP2))

Es donen dues llistes d'igual longitud $A = (x_1, \ldots, x_n)$ i $B = (y_1, \ldots, y_n)$, de mots sobre un cert alfabet Σ, i un nombre natural k, tal que $0 < k < n$. Cada una de les llistes està formada per dues parts $A = (A_1, A_2)$ i $B = (B_1, B_2)$, on $A_1 = (x_1, \ldots, x_k)$, $A_2 = (x_{k+1}, \ldots, x_n)$ $B_1 = (y_1, \ldots, y_k)$, $B_2 = (y_{k+1}, \ldots, y_n)$. Es tracta de determinar si hi ha una seqüència finita no buida d'enters compresos entre 1 i n, $(i_1, \ldots, i_r)$, amb $r \geqslant 1$ tal que $x_{i_1} \cdots x_{i_r} = y_{i_1} \cdots y_{i_r}$, i tal que contingui elements de les dues parts, és a dir que no tots els enters $(i_1, \ldots, i_r)$ estiguin compresos entre 1 i k, ni tots ells ho estiguin entre $k + 1$ i n.

c) (PCP amb tres llistes (PCP3)) Donades tres llistes d'igual longitud $A = (x_1, \ldots, x_n)$, $B = (y_1, \ldots, y_n)$ i $C = (z_1, \ldots, z_n)$, de mots sobre un cert alfabet Σ, determinar si hi ha una seqüència finita no buida d'enters compresos entre 1 i n, $(i_1, \ldots, i_r)$, amb $r \geqslant 1$ tal que $x_{i_1} \cdots x_{i_r} = y_{i_1} \cdots y_{i_r} = z_{i_1} \cdots z_{i_r}$.

Resposta:

a) La següent reducció desde PCP demostra que pcp1 és indecidible.

$$\begin{array}{lll} \langle (u_1, v_1), \ldots, (u_n, v_n) \rangle & \mapsto & \langle (u_1, v_1), \ldots, (u_n, v_n) \rangle \quad \text{, si } n \text{ és senar.} \\ & \mapsto & \langle (u_1, v_1), \ldots, (u_n, v_n), (u_n, v_n) \rangle \quad \text{, si } n \text{ és parell.} \end{array}$$

Efectivament, es tracta d'una funció computable que transforma entrades vàlides de PCP en entrades vàlides de PCP1, i preserva la resposta en ambdós problemes.

Solució alternativa (simplificada):

La següent reducció també funciona.

$$\langle (u_1, v_1), \ldots, (u_n, v_n) \rangle \mapsto \langle (u_1, v_1), \ldots, (u_n, v_n), (u_1, v_1), \ldots, (u_n, v_n), (u_1, v_1) \rangle$$

b) Reduïm novament desde PCP:

$$\langle (u_1, v_1), \ldots, (u_n, v_n) \rangle \mapsto \langle (u_1, v_1), \ldots, (u_n, v_n) || (u_1, v_1), \ldots, (u_n, v_n) \rangle$$

La transformació és computable. Si el problema original té resposta afirmativa amb $(i_1, \ldots, i_k)$ en PCP, llavors la seva imatge té resposta afirmativa amb $(i_1, \ldots, i_k, i_1 + n, \ldots, i_k + n)$ en PCP2. Vice-versa, si el problema imatge té resposta afirmativa amb $(i_1, \ldots, i_k)$ en PCP2, llavors el problema original té resposta afirmativa amb $(i_1 \ mod \ n, \ldots, i_k \ mod \ n)$, on interpretem $i \ mod \ n$ com el representant dins $\{1, \ldots, n\}$ del conjunt de naturals equivalents a i mòdul n.

Solució alternativa (simplificada):

La següent reducció també funciona.

$$\langle (u_1, v_1), \ldots, (u_n, v_n) \rangle \mapsto \langle (u_1, v_1), \ldots, (u_n, v_n) || (\$, \$) \rangle$$

On $ és un símbol nou.

c) Reduïm novament desde PCP:

$$\langle (u_1, v_1), \ldots, (u_n, v_n) \rangle \mapsto \langle (u_1, v_1, v_1), \ldots, (u_n, v_n, v_n) \rangle$$

És una transformació computable, i es preserva la resposta, ja que $u_{i_1} \ldots u_{i_k} = v_{i_1} \ldots v_{i_k}$ si i només si $u_{i_1} \ldots u_{i_k} = v_{i_1} \ldots v_{i_k}$ i $v_{i_1} \ldots v_{i_k} = v_{i_1} \ldots v_{i_k}$.

Problema 2 Demostreu que un llenguatge és semidecidible si i només si és la imatge d'alguna funció computable (no necessàriament total).

Resposta: Per a la implicació d'esquerra a dreta, suposem que L és semidecidible. Llavors existeix una màquina M que accepta una entrada x qualsevol si i només si x pertany a L. Utilitzant M construïm M' de la manera següent:

```
entrada x
  simular M(x)
  si M accepta x
    llavors sortida x
    si no divergir.
```

Aquest nou programa computa la funció

$$f(x) = \begin{cases} x & \text{si } x \in L \\ \text{indefinida,} & \text{altrament.} \end{cases}$$

Per tant, L és exactament la imatge d'aquesta funció, que al seu temps és computable perquè hi ha una màquina M' que la computa.

Per a la implicació de dreta a esquerra, suposem que L és la imatge d'alguna funció computable f. Com que f és computable, existeix una màquina M que la computa, i llavors per tot y es compleix $y \in L$ si i només si existeix x tal que $M(x) = y$. Construïm la següent màquina M' que accepta l'entrada y si i només si $y \in L$:

```
entrada y
  p:=1 ; trobat:=fals
  mentre no trobat fer
    des de x:=0 fins p fer
      simular M(x) durant p passos
      si s'ha aturat i dóna resultat y
        llavors trobat:=cert
    p:=p+1
  fimentre
  acceptar
```

Problema 3 Trobeu successivament gramàtiques per als tres llenguatges següents.

- $\{ww^{\mathsf{R}} \in \{0,1\}^* \mid \text{valor}_2(w) = \dot{3}\}$
- $\{ww^{\mathsf{R}} \in \{0,1\}^* \mid \text{valor}_2(w^{\mathsf{R}}) = \dot{3}\}$
- $\{ww^{\mathsf{R}} \in \{0,1\}^* \mid \text{valor}_2(ww^{\mathsf{R}}) = \dot{3}\}$

Resposta:

- La gramàtica, essencialment, combinarà la gramàtica per $\{ww^R\}$ amb l'autòmat dels múltiples de 3.

	0	1
$\dagger q_0$	q_0	q_1
q_1	q_2	q_0
q_2	q_1	q_2

$$\begin{array}{lcllll} S_0 & \to & 0S_00 & \mid & 1S_11 & \mid \quad \lambda \\ S_1 & \to & 0S_20 & \mid & 1S_01 & \\ S_2 & \to & 0S_10 & \mid & 1S_21 & \end{array}$$

- L'autòmat anterior no varia si se li aplica la operació de revessat. Això implica que $\mathtt{valor}_2(w)$ és múltiple de 3 si i només si $\mathtt{valor}_2(w^R)$ és múltiple de 3. Per tant, la gramàtica resultant és la mateixa que la de l'apartat anterior.

- Per la forma de l'autòmat dels múltiples de 3, si desde qualsevol estat q llegim dos cops el mateix símbol a, ens quedem en el propi estat q; és a dir, $qaa = q$. Per tant, donada qualsevol paraula $w = a_1 \dots a_k$, resulta que $q_0ww^R = q_0(a_1 \dots a_k a_k \dots a_1) = (q_0a_1 \dots a_{k-1})a_ka_k(a_{k-1} \dots a_1) = (q_0a_1 \dots a_{k-1})(a_{k-1} \dots a_1) = (q_0a_1 \dots a_{k-2})a_{k-1}a_{k-1}(a_{k-2} \dots a_1) = (q_0a_1 \dots a_{k-2})(a_{k-2} \dots a_1) = \dots = q_0$ Així doncs, qualsevol paraula ww^R és múltiple de 3, i per tant una gramàtica que accepti el tercer llenguatge podria ser la següent.

$$S \quad \to \quad 0S0 \quad \mid \quad 1S1 \quad \mid \quad \lambda$$

Problema 4 Demostreu que no és regular el llenguatge format per tots els mots sobre l'alfabet $\{a\,,\,b\}$ tals que, si la seva longitud és múltiple de tres, llavors han de ser de la forma a^nb^n; formalment:

$$\{w \in \{a\,,\,b\}^* \mid (\|w\| = \dot{3}) \implies (\exists n\; w = a^nb^n)\}.$$

Resposta: Ho demostrem pel lema de bombament. Fixat N, escollim la paraula $a^{3N}b^{3N}$, que és del llenguatge. Sigui $w = xyz$ una factorització qualsevol cumplint $\|xy\| \leqslant N$ i $\|y\| \geqslant 1$. Necessàriament, x, y, z són de la forma a^j, a^k, $a^{3N-j-k}b^{3N}$, respectivament, amb $k \geqslant 1$. La paraula xy^4z té longitud $3N + 3N + 3k$, que és múltiple de 3. En canvi, no és de la forma a^nb^n, doncs $3N + 3k > 3N$. Per tant, no és del llenguatge, i amb això finalitza la prova.

Solució alternativa:

Suposem que el llenguatge L és regular. Sigui $\sigma : \{a, b\}^* \to \{a, b\}^*$ el morfisme definit per $\sigma(a) = aaa$ i $\sigma(b) = bbb$. Com que els llenguatges regulars són tancats per morfisme invers, resulta que $\sigma^{-1}(L) = \{a^nb^n\}$ és regular també; però aquest és el classic llenguatge conegut com a no regular: contradicció.

Solució alternativa:

Suposem que el llenguatge L és regular. Per simplicitat, abusem i anomenem $\dot{3}$ al llenguatge de les paraules de longitud múltiple de tres, que és regular. Com que els llenguatges regulars són tancats per intersecció, resulta que $L \cap \dot{3}$ també és regular. Però $L \cap \dot{3}$ és $\{a^{3n}b^{3n}\}$, y com que els regulars són tancats per concatenació i unió, resulta que $\{a^nb^n\} = \{a^{3n}b^{3n}\} \;\cup\; a\{a^{3n}b^{3n}\}b \;\cup\; aa\{a^{3n}b^{3n}\}bb$ també és regular; però aquest és el classic llenguatge conegut com a no regular: contradicció.

Problema 5 Construïu el DFA mínim que accepta els mots que pertanyen a la intersecció del llenguatge generat per la gramàtica

$$S \to B \mid \lambda$$
$$B \to aS \mid bB$$

i del llenguatge definit per l'expressió regular $\left(a + b\,(ab^*a)^*\,b\right)^*$.

Resposta (simplificada): Reemplaçant la primera B per les seves corresponents parts dretes ens deixa la següent gramática.

$$\begin{array}{lllllll} S & \to & aS & \mid & bB & \mid & \lambda \\ B & \to & aS & \mid & bB & & \end{array}$$

Aquesta gramática és lineal, i per tant accepta un llenguatge regular. La simple observació ja mostra que genera totes les paraules que no acaben en b. De totes maneres, la traducció a autòmat és mecànica.

	a	b
$\dagger q_0$	q_0	q_1
q_1	q_0	q_1

L'expressió regular és la dels múltiples de 3, on s'han intercanviat 0's per a's i 1's per b's. Per tant escrivim directament l'autòmat, que de totes maneres es podria obtenir mecànicament aplicant operacions de tancament.

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

La intersecció dels dos autòmats, juntament amb el següent procés de minimització, dona lloc finalment a l'autòmat que descrivim a continuació.

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

Problema 6 Construïu el DFA mínim que reconeix el llenguatge format pels mots sobre l'alfabet $\{a\,,\,b\}$ tals que tot sufix que conté un nombre parell de a's també conté un nombre parell de b's.

Resposta (simplificada):

Una possible formalització d'aquest llenguatge seria la següent.

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

La negació del llenguatge anterior dona lloc a

$$\{w \in \{a,b\}^* \mid \exists x,y : (w = xy \wedge |y|_a = \dot{2} \wedge |y|_b \neq \dot{2})\}$$

I aquest llenguatge es pot posar com la següent concatenació.

$$\{a,b\}^*\{y \in \{a,b\}^* \mid |y|_a = \dot{2} \wedge |y|_b \neq \dot{2})\}$$

Tots dos llenguatges tenen autòmats directes. Si els construïm i anem aplicant les corresponents operacions de tancament, determinització, minimització i complementari, obtindrem finalment

	a	b
$\dagger q_0$	q_1	q_3
$\dagger q_1$	q_1	q_2
q_2	q_6	q_5
q_3	q_4	q_3
$\dagger q_4$	q_5	q_6
q_5	q_4	q_2
q_6	q_6	q_6

Solució alternativa (simplificada):

El llenguatge revessat és el de les paraules tals que tot prefix amb un nombre parell de a's té també un nombre parell de b's. Aquest llenguatge té un autòmat que es pot obtenir directament com segueix (recordant cada estat si hem vist parell de a's o de b's fins al moment).

	a	b
$\dagger q_{2a2b}$	q_{1a2b}	q_{2a1b}
$\dagger q_{1a2b}$	q_{2a2b}	q_{1a1b}
q_{2a1b}	q_{1a1b}	q_{2a2b}
$\dagger q_{1a1b}$	q_{2a1b}	q_{1a2b}

Si li apliquem el revessat, determinitzem i minimitzem, obtindrem l'autòmat de la solució anterior.

(Examen de Gener-2006)

Problema 1 Demostreu que les dues variants següents del problema dels mots també són indecidibles.

a) (WP amb pas intermedi (WP1))

Donada una llista finita R de regles de reescriptura (un conjunt finit de parells de mots (u_i, v_i)) i tres mots u (inicial), v (intermedi) i w (final), determinar si és possible passar de u a v i de v a w utilitzant les regles de reescriptura (un seguit de substitucions d'algun submot u_i pel seu corresponent v_i). Formalment, el llenguatge associat a aquest problema és el següent:

$$\{\langle R, u, v, w\rangle \mid u \overset{*}{\to} v \wedge v \overset{*}{\to} w\}.$$

b) (WP amb confluència (WP2)) Donada una llista finita R de regles de reescriptura (un conjunt finit de parells de mots (u_i, v_i)) i dos mots *inicials* u' i u'', determinar si existeix algun mot final comú v tal que sigui possible passar de u' a v i de u'' a v utilitzant les regles de reescriptura (un seguit de substitucions d'algun submot u_i pel seu corresponent v_i). Formalment, el llenguatge associat a aquest problema és el següent:

$$\{\langle R, u', u''\rangle \mid \exists v\ (u' \overset{*}{\to} v \wedge u'' \overset{*}{\to} v)\}.$$

Resposta: La transformació $f(\langle R, u, v\rangle) = \langle R, u, u, v\rangle$ redueix WP a WP1. La prova és evident.

La transformació $g(\langle R, u, v\rangle) = \langle S, u, \$\rangle$, on $S = R \cup \{v \to \$\}$ i \$ és un nou símbol, redueix WP a WP2.

Si $u \to_R^* v$, llavors $u \to_S^* \$$, de manera que u i \$ conflueixen amb S. Vice-versa, si u i \$ conflueixen amb S, llavors, donat que \$ no és reescrivible, resulta que $u \to_S^* \$$. Aquesta derivació tindrá un únic pas amb la regla $v \to \$$, doncs \$ apareix al final, i cap regla de $S = R \cup \{v \to \$\}$ esborra \$, i per tant és de la forma $u \to_R^* u_1 v u_2 \to_{v\to\$} u_1 \$ u_2 \to_R^* \$$, de manera que hi apareixen dues subderivacions $u_1 \to_R^* \lambda$ i $u_2 \to_R^* \lambda$. De tot plegat en resulta $u \to_R^* u_1 v u_2 \to_R^* \lambda v \lambda = v$.

Problema 2 Siguin x_1 i x_2 dues entrades donades diferents qualssevol. Classifiqueu com a decidible, no decidible però semidecidible, o no semidecidible el conjunt següent:

$$\{x \mid \mathrm{M}_x(x_1)\downarrow \;\wedge\; \mathrm{M}_x(x_2)\downarrow \;\wedge\; \mathrm{M}_x(x_1)\neq\mathrm{M}_x(x_2)\}$$

Resposta: El problema de $\mathcal{K}$ es redueix al nostre segons la transformació que, donat un `x` d'entrada, dona com a resultat el número associat al programa següent:

```
entrada y
   Simular M_x(x)
   sortida y
```

És evident que aquest programa s'atura, i dona sortida diferent amb entrades x_1 i x_2 diferents, si i només si $x \in \mathcal{K}$. Així doncs, el problema de l'enunciat és indecidible. Però es pot semi-decidir amb el següent procés.

```
entrada x
   Simular M_x(x_1)
   Simular M_x(x_2)
   Si totes dues execucions aturen i donen sortida diferent llavors acceptar
   Si no rebutjar
```

Problema 3 Construïu una gramàtica, preferiblement no ambigua, que generi tots els mots sobre l'alfabet $\{a, b, c\}$ de la forma w_1cw_2 on w_1 i w_2 no contenen cap c, tenen la mateixa longitud i són tals que si w_1 conté dues a's seguides, llavors w_2 conté dues b's seguides. Formalment,

$$\{w_1cw_2 \mid w_1, w_2 \in \{a\,,\,b\}^* \wedge \|w_1\| = \|w_2\| \wedge (w_1 \in \{a\,,\,b\}^*aa\{a\,,\,b\}^* \implies w_2 \in \{a\,,\,b\}^*bb\{a\,,\,b\}^*)\}.$$

Resposta: La següent gramàtica genera el llenguatge demanat.

$$\begin{array}{lcccccccccc}
S & \to & aS_aa & \mid & aS_{ab}b & \mid & bSa & \mid & bS_bb & \mid & c \\
S_a & \to & aS_{aa}a & \mid & aS_{aab}b & \mid & bSa & \mid & bS_bb & \mid & c \\
S_{ab} & \to & aS_{aa}a & \mid & aS_{bb}b & \mid & bSa & \mid & bS_{bb}b & \mid & c \\
S_{aa} & \to & aS_{aa}a & \mid & aS_{aab}b & \mid & bS_{aa}a & \mid & bS_{aab}b & & \\
S_{aab} & \to & aS_{aa}a & \mid & aS_{bb}b & \mid & bS_{aa}a & \mid & bS_{bb}b & & \\
S_b & \to & aS_aa & \mid & aS_{bb}b & \mid & bSa & \mid & bS_{bb}b & \mid & c \\
S_{bb} & \to & aS_{bb}a & \mid & aS_{bb}b & \mid & bS_{bb}a & \mid & bS_{bb}b & \mid & c
\end{array}$$

Es pot demostrar fàcilment (tot i que tediosament) veient abans, per inducció sobre el nombre de reescriptures i sobre la longitud de les paraules, els següents fets:

- $S \to^* uSv$ si i només si $u, v \in \{a, b\}^*$, $\|u\| = \|v\|$, u no conté dues a's seguides, v no conté dues b's seguides, $u \neq u'a$, $v \neq bv'$.
- $S \to^* uS_av$ si i només si $u, v \in \{a, b\}^*$, $\|u\| = \|v\|$, u no conté dues a's seguides, v no conté dues b's seguides, $u = u'a$, $v \neq bv'$.

- $S \to^* uS_{ab}v$ si i només si $u, v \in \{a, b\}^*$, $\|u\| = \|v\|$, u no conté dues a's seguides, v no conté dues b's seguides, $u = u'a$, $v = bv'$.
- $S \to^* uS_{aa}v$ si i només si $u, v \in \{a, b\}^*$, $\|u\| = \|v\|$, u conté dues a's seguides, v no conté dues b's seguides, $v \neq bv'$.
- $S \to^* uS_{aab}v$ si i només si $u, v \in \{a, b\}^*$, $\|u\| = \|v\|$, u conté dues a's seguides, v no conté dues b's seguides, $v = bv'$.
- $S \to^* uS_{b}v$ si i només si $u, v \in \{a, b\}^*$, $\|u\| = \|v\|$, u no conté dues a's seguides, v no conté dues b's seguides, $v = bv'$.
- $S \to^* uS_{bb}v$ si i només si $u, v \in \{a, b\}^*$, $\|u\| = \|v\|$, v conté dues b's seguides.

Problema 4 Demostreu que el llenguatge $\{a^r b^s \mid r, s \in \mathbb{N} \ \wedge \ 9r \leqslant s \leqslant 10r\}$ no és regular.

Resposta: Ho demostrem utilitzant el lema de bombament. Fixat N, escollim $w = a^N b^{9N}$, que és del llenguatge. Sigui $w = xyz$ una factorització qualsevol tal que $\|xy\| \leqslant N$ i $\|y\| \geqslant 1$. Llavors x, y i z són, respectivament, de la forma a^j, a^k i $a^{N-j-k}b^{9N}$. La paraula xy^2z no és del llenguatge, doncs $9N + k > 9N$.

Problema 5 Construïu el DFA mínim que accepta només els mots sobre l'alfabet $\{a\,,\, b\}$ tals que tot submot bab va precedit immediatament o seguit immediatament per almenys una a.

Resposta (simplificada): Formalitzem el llenguatge de la següent manera.

$$L \;=\; \{w \mid \forall x, y : (w = xbaby \Rightarrow \exists u : (x = ua \vee y = au))\}$$

I passant al complementari.

$$\overline{L} \;=\; \{w \mid \exists x, y : (w = xbaby \wedge \forall u : (x \neq ua \wedge y \neq au))\}$$

Així que podem posar el llenguatge com la següent concatenació.

$$\overline{L} \;=\; \overline{(\Sigma^* a)}bab\overline{(a\Sigma^*)}$$

Tots els subllenguatges que apareixen a la expressió tenen autòmats bàsics. Ara caldria construir-los i anar aplicant després les corresponents operacions de tancament fins a obtenir l'autòmat pel llenguatge original, que resulta en el següent.

	a	b
$\dagger q_1$	q_2	q_3
$\dagger q_2$	q_2	q_1
$\dagger q_3$	q_4	q_3
$\dagger q_4$	q_2	q_5
q_5	q_2	q_6
q_6	q_6	q_6

(Examen de Juny-2006)

Problema 1 (2 punts en total) Considerem les dues variants següents del problema dels mots. Justifiqueu que la primera és decidible (per exemple, explicant un algorisme que la resol) i que la segona no ho és (per exemple, reduïnt des del problema dels mots).

1. ***WP* amb mots curts (wpc)** (0.75) Donada una llista finita R de regles de reescriptura (un conjunt finit de parells de mots (u_i, v_i)) i dos mots u (inicial) i v (final), determinar si és possible passar de u a v utilitzant les regles de reescriptura (un seguit de substitucions d'algun submot u_i pel seu corresponent v_i), tenint, en tot punt de la derivació, cadenes intermèdies de mida més petita o igual a $|v|$. Formalment, el llenguatge associat a aquest problema és el següent:

$$\{\langle R, u, v\rangle \mid \exists k \geqslant 1\ \exists w_1, \dots w_k : \quad (u = w_1 \to_R w_2 \to_R w_3 \dots \to_R w_k = v \quad \wedge \quad \forall i \in \{1 \dots k\} : (|v| \geqslant |w_i|))\}$$

2. ***WP* amb mots llargs (wpl)** (1.25) Donada una llista finita R de regles de reescriptura (un conjunt finit de parells de mots (u_i, v_i)) i dos mots u (inicial) i v (final), determinar si és possible passar de u a v utilitzant les regles de reescriptura (un seguit de substitucions d'algun submot u_i pel seu corresponent v_i), tenint, en algun punt de la derivació, una cadena intermèdia més gran que $|v|$. Formalment, el llenguatge associat a aquest problema és el següent:

$$\{\langle R, u, v\rangle \mid \exists k \geqslant 1\ \exists w_1, \dots w_k : \quad (u = w_1 \to_R w_2 \to_R w_3 \dots \to_R w_k = v \quad \wedge \quad \exists i \in \{1 \dots k\} : (|v| < |w_i|))\}$$

Resposta:

1. La idea és generar totes les paraules accessibles desde u amb una derivació contenint només paraules de mida menor o igual a $|v|$, i comprovar si v està entre elles. Això és pot fer en temps finit perquè d'aquestes n'hi ha un nombre finit, i s'hi pot arribar a totes elles segons les condicions del problema a través d'aquest mateix conjunt finit, sense utilitzar cap paraula que no hi pertanyi. N'hi ha prou amb anar reescrivint desde u de totes les maneres possibles, quedant-nos només amb les paraules de mida menor o igual que $|v|$, així, fins que no n'apareguin de noves. Finalment, si v ha aparegut entre elles acceptem, i si no, rebutjem.

2. Reduïm WP a WPL amb la següent transformació.

$$\langle u_1 \to v_1, \dots, u_n \to v_n, u, v\rangle \mapsto \langle u_1 \to v_1, \dots, u_n \to v_n, \$ \to \lambda,\ \$^{|v|+1}u,\ v\rangle$$

On \$ s'ha escollit com a símbol nou que no apareixia en el sistema inicial ni les paraules inicials.

Sigui R el sistema de partida. Si $u \to_R^* v$, llavors està clar que $\$^{|v|+1}u \to_{R\cup\{\$\to\lambda\}}^* v$. Además, com que $|\$^{|v|+1}u| > |v|$, està clar que en aquesta derivació hi apareixen paraules de mida major estricte que la de v.

Per la direcció contraria, suposem que $\$^{|v|+1}u \to_{R\cup\{\$\to\lambda\}}^* v$ amb paraules llargues. Com que en aquesta derivació la regla $\$ \to \lambda$ tracta exclusivament als símbols \$, hi ha una subderivació $u \to_R^* v$.

Resposta alternativa al primer apartat:
L'algorisme següent decideix aquest conjunt:

```
entrada ⟨R, u, v⟩
S:={u}
S_anterior:=∅
Mentre S ≠ S_anterior:
   S_anterior:=S
   Per cada w ∈ S:
      Per cada t arribable desde w en un pas de reescriptura i tal que |w| ⩽ v:
         S:=S ∪ {t}
Si v ∈ S Llavors acceptar
Sino rebutjar
```

Al conjunt S s'hi acaben afegint totes les paraules w de mida menor o igual a $|v|$ a les quals s'hi pot arribar desde u passant només per cadenes intermèdies de mida menor o igual a $|v|$. Això es demostraria per inducció sobre el nombre de voltes del bucle principal de l'algorisme, i per inducció sobre el nombre de passos de reescriptura.

Resposta alternativa al segon apartat: Donat que WP=WPC $\cup$ WPL, i que ja hem vist a l'apartat anterior que WPC és decidible, resulta que WPL no pot ser decidible: si ho fos, com que la unió de decidibles és decidible, implicaria que WP també ho sería, cosa que ja sabem que no és així.

Problema 2 Formalitzeu i classifiqueu com a decidibles, no decidibles però semidecidibles, o no semidecidibles els conjunts següents:

1. (1) Conjunt de programes tals que s'aturen per a almenys dues entrades.
2. (0.75) Conjunt de programes tals que s'aturen per a com a molt una entrada.
3. (0.75) Conjunt de programes tals que s'aturen per a almenys dues entrades, i no s'aturen per a almenys dues entrades.

Resposta:

1. El conjunt queda definit com segueix.

$$C = \{x \mid \exists y, z : (y \neq z \wedge M_x(y)\downarrow \wedge M_x(z)\downarrow)\}$$

El següent procés semidecideix aquest conjunt.

```
entrada x
t:=1
Executar indefinidament:
   comptador:=0
   Desde i:=0 fins t fer
      Simular Mx(i) durant t passos
      Si la execució atura llavors comptador:=comptador+1
   Si comptador≥2 llavors acceptar
   t:=t+1
```

Per a demostrar que no és decidible, es pot reduïr desde K de la següent manera. Si x és l'element d'entrada, donem com a sortida el programa p següent:

```
entrada y
   Simular Mx(x)
   acceptar
```

Està clar que si x pertany a K, llavors p s'atura amb almenys dues entrades, doncs de fet s'atura sempre. Si x no pertany a K, llavors p no s'atura amb almenys dues entrades, doncs de fet no s'atura mai.

Així doncs, aquest conjunt no és decidible, però sí semi-decidible.

2. El conjunt queda definit com segueix.

$$C' = \{x \mid \forall y, z : (y \neq z \Rightarrow M_x(y) \uparrow \vee M_x(z) \uparrow)\}$$

Donat que és complementari de l'anterior, i que els decidibles són tancats per complementari, aquest tampoc pot ser decidible. Además, com que un llenguatge no decidible i el seu complementari no poden ser semi-decidibles alhora, tenint en compte que l'anterior sí era semi-decidible, podem concloure que aquest no és ni semi-decidible.

3. El conjunt queda definit com segueix.

$$C'' = \{x \mid \exists y, z : (y \neq z \wedge M_x(y) \downarrow \wedge M_x(z) \downarrow) \wedge \exists y, z : (y \neq z \wedge M_x(y) \uparrow \wedge M_x(z) \uparrow)\}$$

Demostrarem que no és ni semi-decidible reduïnt desde $\overline{\mathtt{K}}$ de la següent manera. Si x és l'element d'entrada, donem com a sortida el programa p següent:

```
entrada y
   Si y=1 o y=2 llavors aturar
   Simular M_x(x)
   aturar
```

Si x pertany a $\overline{\mathtt{K}}$, llavors $\mathrm{M}_x(x) \uparrow$, de manera que el programa p s'atura amb dues entrades (1 i 2) i no s'atura amb tota la resta, que almenys són dues, i per tant p pertany al nostre conjunt C''.

En canvi, si x no pertany a $\overline{\mathtt{K}}$, llavors $\mathrm{M}_x(x) \downarrow$, de manera que el programa p s'atura amb totes les entrades, i per tant p no pertany al nostre conjunt C''.

Problema 3 (1.5 punts en total) Demostreu que el llenguatge $\{ww \mid w \in \{a,b\}^*\}$ no és regular (1 punts). Determineu també si $\{ww \mid w \in a^*\}$ és regular o no (0.5 punts).

Resposta: Justifiquem que no és regular utilitzant el lema de bombament. Agafem un N qualsevol fixat i escollim la paraula $w = a^N b a^N b$, que és del llenguatge. Sigui $w = xyz$ una factorització qualsevol tal que $|xy| \leqslant N$ i $|y| \geqslant 1$. Resulta que el mot xy^2z no pertany al llenguatge.

El conjunt $\{ww \mid w \in a^*\}$ es pot formular també com $\{a^{2n}\}$, és a dir, que és el conjunt de paraules de longitud parell formades amb només a's. Com a expressió regular es pot descriure així: $(aa)^*$. Per tant, és regular.

Problema 4 (4 punts en total) Contesteu a les qüestions següents.

1. (2) Considerem la gramàtica:

$$\begin{array}{lcccccc} S & \to & YaX_0 & & & & \\ X_0 & \to & X_1X_1 & & & & \\ X_1 & \to & a & \mid & b & & \\ Y & \to & aY & \mid & bY & \mid & \lambda \end{array}$$

Doneu una definició formal del llenguatge que genera amb la notació habitual de descripció de conjunts, i justifiqueu que és regular tot trobant l'autòmat mínim que el reconeix.

2. (1) Considerem la generalització de la gramàtica anterior, de la següent manera, per un n qualsevol fixat:

$$
\begin{array}{lclcl}
S & \to & YaX_0 & & \\
X_0 & \to & X_1X_1 & & \\
X_1 & \to & X_2X_2 & & \\
\vdots & & & & \\
X_{n-1} & \to & X_nX_n & & \\
X_n & \to & a & | & b \\
Y & \to & aY & | & bY \quad | \quad \lambda
\end{array}
$$

Doneu una definició formal del llenguatge que genera amb la notació habitual de descripció de conjunts, i argumenteu que és regular.

3. (1) Raoneu quants estats té l'autòmat més petit que reconeix el llenguatge de l'apartat anterior.

Resposta:

1. La gramàtica força l'aparició d'una a en la paraula generada, en la tercera posició comptant desde el final, ja que la variable Y genera qualsevol paraula, i la variable X_0 genera qualsevol paraula de longitud 2. Per tant, el llenguatge queda definit com segueix.

$$C = \{w \mid \exists x, y : (w = xay \wedge |y| = 2)\}$$

També el podem expressar en forma d'expressió regular:

$$\{a,b\}^*a\{a,b\}\{a,b\}$$

o més simplificadament com:

$$\{a,b\}^*a\{a,b\}^2$$

Per tant, és regular. Aplicant les operacions de tancament obtenim la representació següent, com a autòmat mínim, del mateix llenguatge, que també es podia obtenir de manera directa amb el següent raonament: només cal recordar els últims 3 símbols vistos per poder saber en tot moment si hi ha una 'a' en la tercera posició abans del final, i, de fet, només cal recordar en quina de les 3 últimes posicions hi ha una 'a'.

	a	b
q_{bbb}	q_{bba}	q_{bbb}
q_{bba}	q_{baa}	q_{bab}
q_{baa}	q_{aaa}	q_{aab}
q_{bab}	q_{aba}	q_{abb}
$\dagger q_{aaa}$	q_{aaa}	q_{aab}
$\dagger q_{aab}$	q_{aba}	q_{abb}
$\dagger q_{aba}$	q_{baa}	q_{bab}
$\dagger q_{abb}$	q_{bba}	q_{bbb}

2. Similarment a l'apartat anterior, la gramàtica força a l'aparició d'una 'a' en una posició concreta de la paraula, deixant a la dreta un sufix de longitud 2^n, ja que desde X_0 s'acaben generant 2^n ocurrències de X_n. Per tant, el llenguatge queda definit com segueix.

$$C = \{w \mid \exists x, y : (w = xay \wedge |y| = 2^n)\}$$

També el podem expressar en forma d'expressió regular:

$$\{a,b\}^* a \overbrace{\{a,b\}\ldots\{a,b\}}^{2^n)}$$

o més simplificadament com

$$\{a,b\}^* a \{a,b\}^{2^n}$$

El sol fet que es pugui representar com a expressió regular ja justifica que el llenguatge és regular.

3. Podriem construïr un autòmat que reconegui aquest llenguatge a base de recordar, en cada estat, en quines de les últimes 2^n+1 lectures hi hem vist una 'a'; i acceptant en el cas en que el de la posició 2^n+1 començant pel final contingui una a. Això requereix 2^{2^n+1} estats.

   Però aquesta idea ja dona l'autòmat mínim, com justifiquem a continuació. En un estat de l'autòmat necessito recordar els últims $\{2^n+1\}$ símbols vistos, perque si me n'oblido d'un d'ells (si és a o b), quan llegeixi uns quants símbols i aquest estigui a distancia 2^n+1 del final, no sabré si acceptar i rebutjar. Recordar 2^n+1 booleans dona lloc a 2^{2^n+1} estats diferents.

Resposta alternativa a l'últim apartat: Podriem construïr un autòmat que reconegui aquest llenguatge a base de recordar, en cada estat, en quines de les últimes 2^n+1 lectures hi hem vist una 'a'; i acceptant en el cas en que el de la posició 2^n+1 començant pel final contingui una a. Això requereix 2^{2^n+1} estats.

Però aquesta idea ja dona l'autòmat mínim, com justifiquem a continuació. En un exercici del curs haviem vist que un llenguatge de la forma $\{a,b\}^*a\{a,b\}^t$ requereix almenys de 2^{t+1} estats. Com que en el nostre cas t és 2^n, resulta que necessitem 2^{2^n+1} estats.

Resposta alternativa a l'últim apartat: Podriem construïr un autòmat que reconegui aquest llenguatge a base de recordar, en cada estat, en quines de les últimes 2^n+1 lectures hi hem vist una 'a'; i acceptant en el cas en que el de la posició 2^n+1 començant pel final contingui una a. Això requereix 2^{2^n+1} estats.

Però aquesta idea ja dona l'autòmat mínim, com justifiquem a continuació. Suposem que existeix un autòmat A que reconeix el llenguatge i amb menys estats que 2^{2^n+1}, i arribarem a una contradicció. Considerem totes les paraules diferents de mida 2^n+1. D'aquestes n'hi ha 2^{2^n+1}. Com que n'hi ha més que estats, dues ens han de portar al mateix estat començant l'execució desde l'estat inicial. Siguin x i y dues paraules diferents de mida 2^n+1 que porten al mateix estat q. Sigui z el prefix comú més llarg de x i y. Donat que x i y són diferents, llavors són de la forma, o bé $x=zax'$ i $y=zby'$, o bé $x=zbx'$ i $y=zay'$. Sense pèrdua de generalitat considerem només el primer cas (el segon es tracta de manera anàloga), i per tant $x=zax'$ i $y=zby'$. Donat que aquestes dues paraules ens porten al mateix estat, les paraules $zax'a^{2^n-|x'|}$ i $zby'a^{2^n-|x'|}$ (noteu que $|x'|=|y'|$) també ens porten al mateix estat, que serà acceptador o no. Per tant, o bé s'accepta una paraula com $zby'a^{2^n-|x'|}$ que no conté una a en la posició 2^n+1 abans del final, o bé es rebutja una paraula com $zax'a^{2^n-|x'|}$ que sí conté una a en la posició 2^n+1 abans del final: contradicció amb el fet que l'autòmat A reconeix el llenguatge.

(Examen de Gener-2007)

1. (7 punts en total) Considereu els llenguatges següents:

   - $L_1 = \{0^{n_1}10^{n_2}1\ldots 0^{n_{k-1}}10^{n_k} \mid (\forall i : (1 \leqslant i \leqslant k \Rightarrow n_i \geqslant 0)) \wedge$
     $(\exists i, j : (1 \leqslant i < j \leqslant k \ \wedge \ i+2 \neq j \ \wedge \ n_i = n_j))\}$
   - $L_2 = \{0^i10^j10^k \mid i = j \vee j = k\}$
   - $L_3 = \{w_1 1 w_2 1 w_3 \mid w_1, w_2, w_3 \in \{a, b, c\}^* \wedge (|w_1| = |w_2| \vee |w_2| = |w_3|)\}$
   - $L_4 = \{a^i1b^j1c^k \mid i = j \vee j = k\}$

   Responeu raonadament a les qüestions següents. Tingueu en compte que, tot i estar relacionades, es poden resoldre de manera independent. Quan us convingui, podeu suposar que els apartats anteriors han estat ja demostrats. De manera indicativa, la segona i penúltima qüestions són les més difícils.

   (a) (0.5 punts) Quins dels mots següents són de L_1? 0, 1, 010, 01010, 010010, 010010100 (en aquest apartat no cal que justifiqueu la resposta).

   (b) (2 punts) Doneu una gramàtica que generi L_1.

   (c) (0.75 punts) Definiu formalment un llenguatge regular L_2' tal que $L_1 \cap L_2' = L_2$, i trobeu l'autòmat mínim que el reconeix.

   (d) (0.75 punts) Definiu un morfisme σ tal que $\sigma^{-1}(L_2) = L_3$. Amb el σ que heu trobat, quines inclusions es compleixen d'entre $\sigma(L_4) \subseteq L_2$ i $L_2 \subseteq \sigma(L_4)$?

   (e) (0.5 punts) Definiu, amb una expressió regular, un llenguatge regular L_4' tal que $L_3 \cap L_4' = L_4$ (no cal que en trobeu cap autòmat ni que justifiqueu aquest apartat).

   (f) (0.75 punts) Als exercicis del curs vàrem veure que el llenguatge $\{a^ib^jc^k \mid i = j \vee j = k\}$, de definició molt semblant a L_4, era inherentment ambigu. La demostració es pot adaptar molt fàcilment per veure que el propi L_4 també és inherentment ambigu, i per tant, suposem aquest fet com a cert. Utilitzeu això, i el fet que els llenguatges incontextuals no inherentment ambigus són tancats per intersecció amb regulars i per morfisme invers, per justificar que L_1 també és inherentment ambigu.

   (g) (1 punts) Justifiqueu que els llenguatges regulars són no ambigus (en aquesta pregunta no cal una justificació massa detallada; n'hi ha prou amb què doneu una idea de com es podria demostrar).

   (h) (0.75 punts) Demostreu que el llenguatge L_1 no és regular.

Resposta:

(a) Els mots 1, 010, 01010 són de L_1, mentre que els mots 0, 010010, 010010100 no ho són.

(b) Intuïtivament, els mots de L_1 estan formats per grups de 0's separats per 1's. La condició $(\exists i, j : (1 \leqslant i < j \leqslant k \ \wedge \ i+2 \neq j \ \wedge \ n_i = n_j))$ ens demana que hi hagi dos grups de 0's que coincideixin en nombre de 0's, i que no quedi enmig d'ells exactament un altre grup de 0's.

Així doncs, qualsevol mot de L_1 és de la forma $x0^ky0^kz$, on x, y i z satisfan les condicions següents. El mot x és o bé λ o bé qualsevol mot acabat en 1. El mot z és o bé λ o bé qualsevol mot que comença per 1. El mot y és o bé 1 o bé comença i acaba per 1 i té almenys tres 1's. Així garantitzem que els dos 0^k no estan separats per exactament un grup de 0's.

Per tant, una possible gramàtica per a L_1 és la següent:

$$
\begin{array}{lcllllll}
S & \rightarrow & Y & \mid & A1Y & \mid & Y1A & \mid \quad A1Y1A \\
Y & \rightarrow & 0Y0 & \mid & 1 & \mid & 1A1A1 & \\
A & \rightarrow & 0A & \mid & 1A & \mid & \lambda &
\end{array}
$$

Com a resposta alternativa podem obtenir una gramàtica lineal:

$$
\begin{array}{lcllllll}
S & \rightarrow & S_2 & \mid & X & & \\
X & \rightarrow & 0X & \mid & 1X & \mid & 1S_2 \\
\\
S_2 & \rightarrow & S_3 & \mid & Z & & \\
Z & \rightarrow & Z0 & \mid & Z1 & \mid & S_31 \\
\\
S_3 & \rightarrow & 0S_3' & \mid & 1 & \mid & 1S_4 \\
S_3' & \rightarrow & S_30 & & & & \\
\\
S_4 & \rightarrow & 0S_4 & \mid & 1S_5 & & \\
S_5 & \rightarrow & 0S_5 & \mid & 1S_5 & \mid & 1
\end{array}
$$

(c) El llenguatge $L_2' = 0^*10^*10^*$ intersecat amb L_1 ens escull de L_1 els mots amb exactament tres grups de 0's. La resta de condicions de L_1 garanteixen que almenys dos grups de 0's consecutius coincidiran en nombre de 0's. Per tant $L_1 \cap L_2' = L_2$. Una definició alternativa en notació clàssica de conjunts és $L_2' = \{0^i10^j10^k\}$.

L'autòmat mínim que reconeix aquest llenguatge és prou senzill com per a escriu-re'l directament:

	0	1
q_0	q_0	q_1
q_1	q_1	q_2
$\dagger q_2$	q_2	q_r
q_r	q_r	q_r

És mínim perque el mot λ distingeix a q_2 de tots els altres estats, el mot 1 distingeix a q_0 de q_1, i a q_1 de q_r, i el mot 11 distingeix a q_0 de q_r.

(d) El morfisme σ definit per $\sigma(a) = \sigma(b) = \sigma(c) = 0$ i $\sigma(1) = 1$ satisfà $\sigma^{-1}(L_2) = L_3$ com justifiquem a continuació.

Tot mot de L_3 és de la forma $w_11w_21w_3$ amb $w_1, w_2, w_3 \in \{a, b, c\}^*$, i té com a imatge $0^{|w_1|}10^{|w_2|}10^{|w_3|}$ per σ. A més, cumpleix la condició $(|w_1| = |w_2| \vee |w_2| = |w_3|)\}$, que és precisament la que es requereix per tal que aquesta imatge sigui de L_2; així doncs, $L_3 \subseteq \sigma^{-1}(L_2)$.

Per altra banda, qualsevol mot w de $\sigma^{-1}(L_2)$ té com a imatge un mot $0^i10^j10^k$ de L_2 cumplint en particular $i = j \vee j = k$. Donat que σ preserva els 1's i no n'afegeix de nous, w ha de ser de la forma $w_11w_21w_3$ on $\sigma(w_1) = 0^i$, $\sigma(w_2) = 0^j$ i $\sigma(w_3) = 0^k$. Per la definició de σ, $w_1, w_2, w_3 \in \{a, b, c\}^*$, i a més $(|w_1| = |w_2| \vee |w_2| = |w_3|)\}$, de manera que $w \in L_3$.

Finalment, qualsevol paraula de la forma $a^i1b^j1c^k$ es transforma en $0^i10^j10^k$. Com que a més a més tenim la mateixa condició $i = j \vee j = k$ tant en L_2 com en L_4, resulta que $\sigma(L_4) = L_2$.

(e) El llenguatge $L_4' = a^*1b^*1c^*$ intersecat amb L_3 ens escull de L_3 les paraules de L_4.

(f) Ho fem per reducció a l'absurd. Suposem que L_1 no és inherentment ambigu. Per l'apartat 1b sabem que L_1 és incontextual. Donat que els incontextuals no inherentment ambigus són tancats per intersecció amb regulars, per l'apartat 1c resulta que L_2 és incontextual i no inherentment ambigu. Donat que els incontextuals no inherentment ambigus són tancats per morfisme invers, per l'apartat 1d resulta que

L_3 és incontextual i no inherentment ambigu. Donat que els incontextuals no inherentment ambigus són tancats per intersecció amb regulars, per l'apartat 1e resulta que L_4 és incontextual i no inherentment ambigu. Això últim contradiu el fet que, com es suposava en aquest apartat, L_4 és inherentment ambigu.

(g) **Els llenguatges regulars es poden reconèixer amb autòmats deterministes. Els autòmats deterministes poden ser simulats per una gramàtica lineal. La transformació usual ja assegura que a tota execució amb una paraula d'entrada acceptada per l'autòmat li correspon una i només una derivació amb la gramàtica. També assegura que la gramàtica no genera més paraules que les reconegudes per l'autòmat. Tot plegat implica que la corresponent gramàtica lineal és no ambigua. (Per a més informació consulteu la pàgina 143 dels apunts de llenguatges incontextuals i regulars de la assignatura).**

(h) **Per l'apartat 1f tenim que L_1 és inherentment ambigu. Per l'apartat 1g sabem que tot llenguatge regular és no inherentment ambigu. Tot plegat implica que L_1 no és regular.**

Com a resposta alternativa ho podem fer pel lema de bombament. Fixat N, escollim la paraula 0^N10^N que és de L_1. Qualsevol subparaula xyz cumplint $|xy| \leqslant N$ i $|y| \geqslant 1$ satisfà que x és de la forma 0^i, que y és de la forma 0^j amb $j \geqslant 1$, i que z és de la forma $0^{N-i-j}10^N$. Llavors, la paraula xy^2z no és de L_1.

2. (1.5 punts) Classifiqueu com a decidible, indecidible però semi-decidible, o no semi-decidible, el llenguatge $L = \{x \mid \forall y : (\mathsf{M}_x(y) \downarrow \Rightarrow \mathsf{M}_x(y) = 1)\}$.

Resposta:

Pel teorema de Rice, donat que L és un conjunt d'índexos, que no és ni el total ni el buit, i que conté els nombres que representen la funció totalment indefinida, resulta que L no és ni semi-decidible.

Més en detall, L és un conjunt d'indexos perque en la condició, sempre que apareix x, ho fa amb el contexte M_x, és a dir, és una condició només sobre el comportament del programa M_x, independentment de la seva implementació. L no és buit perque conté, per exemple, a qualsevol nombre de programa que sempre dona sortida 1. L no és el total perque no conté per exemple als nombres de programes que donen sortida 2 per a alguna entrada.

Com a resposta alternativa podem fer la següent reducció des de $\overline{\mathtt{K}}$. Si x és l'element d'entrada, donem com a sortida el programa p següent:

```
entrada y
   Simular Mx(x)
   sortida 2
```

Si x pertany a $\overline{\mathtt{K}}$, llavors $\mathrm{M}_x(x) \uparrow$, de manera que el programa p no s'atura, i llavors la implicació $(\mathsf{M}_x(y) \downarrow \Rightarrow \mathsf{M}_x(y) = 1)$ és certa per tot y perque l'antecedent és fals per tot y, de manera que p pertany a L.

En canvi, si x no pertany a $\overline{\mathtt{K}}$, llavors $\mathrm{M}_x(x) \downarrow$, de manera que el programa p s'atura amb totes les entrades i dóna sortida 2, així que la implicació $(\mathsf{M}_x(y) \downarrow \Rightarrow \mathsf{M}_x(y) = 1)$ és falsa per tot y i, per tant, p no pertany a L.

3. (1.5 punts) L'objectiu d'aquest problema és demostrar que el llenguatge que descrivim a continuació és indecidible. Donada una llista finita R de regles de reescriptura (un conjunt finit de parells de mots (u_i, v_i)) i un mot u (inicial i final), determinar si és possible passar de u a u aplicant una o més regles de reescriptura (un seguit d'una o més substitucions d'algun submot u_i pel seu corresponent v_i). Formalment, el llenguatge associat a aquest problema és el següent:

$$\{\langle R, u\rangle \mid \exists k \geqslant 2\ \exists w_1, \ldots w_k : (u = w_1 \underset{R}{\to} w_2 \underset{R}{\to} w_3 \ldots \underset{R}{\to} w_k = u)\}$$

(a) (0.75) Justifiqueu primer que la reducció $\langle R, u, v\rangle \mapsto \langle R \cup \{v \to u\}, u\rangle$ des del problema de `Mots` no funciona, mitjançant un contraexemple. (És a dir, que no és cert que una entrada del problema dels mots té resposta afirmativa si i només si la seva transformada en una entrada del problema que hem enunciat també en té).

(b) (0.75) Doneu una nova reducció des del problema de `Mots` i justifiqueu que és correcta.

Resposta:

(a) Hi ha moltes maneres de respondre a aquest apartat. Per exemple, la transformació anterior aplicada sobre $\langle\{a \to aaa\}, a, aa\rangle$ donaria lloc a $\langle\{a \to aaa, aa \to a\}, a\rangle$. El primer té resposta negativa per al problema dels mots, mentre que el segon té resposta afirmativa per al llenguatge que s'ha descrit aquí.

O també, i més senzill, sobre $\langle\{a \to a\}, a, aa\rangle$ s'obtindria $\langle\{a \to a, aa \to a\}, a\rangle$ obtenint resposta negativa en el primer cas i positiva en el segon.

(b) Una possible reducció seria: $\langle R, u, v\rangle \mapsto \langle R', \#\rangle$, amb R' definit com $R \cup \{\# \to \$u\$\ ,\ \$v\$ \to \#\}$, on $\#$ i $\$$ són símbols nous (que no apareixen a R ni u ni v).
Si $\langle R, u, v\rangle$ és del problema de `Mots`, llavors $u \xrightarrow[R]{*} v$, i també $\# \xrightarrow[\# \to \$u\$]{} \$u\$ \xrightarrow[R]{*} \$v\$ \xrightarrow[\$v\$ \to \#]{} \#$, de manera que $\langle R', \#\rangle$ satisfà la condició de l'enunciat.
Per altra banda, si $\langle R', \#\rangle$ satisfà la condició de l'enunciat, llavors $\# \xrightarrow[R']{+} \#$. Escollim aquí la derivació més curta de la forma $\# \xrightarrow[R']{+} \#$, de manera que cap paraula intermedia és $\#$. Per les regles que tenim al sistema de rescriptura, aquesta derivació ha de ser de la forma $\# \xrightarrow[\# \to \$u\$]{} \$u\$ \xrightarrow[R']{*} \$v\$ \xrightarrow[\$v\$ \to \#]{} \#$. Es pot veure inductivament que en la derivació intermedia $\$u\$ \xrightarrow[R']{*} \$v\$$ totes les paraules són de la forma $\$w\$$, on w no conté $\#$ ni $\$$: si en un cert punt abans del final es cumpleix la condició, llavors no es pot fer servir $\# \to \$u\$$ per a obtenir la següent paraula, però tampoc $\$v\$ \to \#$, ja que si no $\#$ seria paraula intermedia. De l'anterior també es dedueix que totes les regles que s'apliquen en aquesta subderivació són de R. Per tant $\$u\$ \xrightarrow[R]{*} \$v\$$, i com que les regles de R no contenen $\$$, tambe tenim que $u \xrightarrow[R]{*} v$, de manera que $\langle R, u, v\rangle$ és del problema de `Mots`.

(Examen de Juny-2007)

Tingueu present en tots els exercicis que les respostes s'han de justificar (a menys que es digui el contrari). La mera presentació d'una proposta de resultat no és suficient.

1. (4.5 punts en total) Els següents apartats, tot i estar relacionats, es poden contestar de forma independent, però recomanem començar pel primer, ja que dóna intuició per als tres darrers.

   a) (0.5) Considerem el llenguatge sobre $\{a, b\}^*$ dels mots que acaben en $aaaa$. Escriviu diréctament l'autòmat determinista mínim que el genera (en aquest cas no cal que poseu cap justificació).

   b) (1.5) Donat un nombre $k \geqslant 0$, el llenguatge dels mots que acaben en k a's, $\{a, b\}^*a^k$, és regular. Justifiqueu que hi ha un autòmat determinista amb $k + 1$ estats que el reconeix.

   c) (1.5) Demostreu que no hi ha cap autòmat determinista amb menys que $k+1$ estats que reconegui $\{a, b\}^*a^k$, o, equivalentment, que el de l'apartat anterior és mínim.

   d) (1) Demostreu que el llenguatge dels mots tals que la segona meitat d'ells conté només a's, és a dir $\{wa^{|w|} | w \in \{a, b\}$, no és regular.

Resposta:

a)

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

b) Generalitzant la idea de l'apartat anterior, només cal que recordem quantes a's hem vist fins al moment. A l'estat q_i hi arribem amb paraules que tenen i a's al final i no $i + 1$ a's al final, excepte per al cas de q_k, que hi arribem amb paraules que tenen k o més a's al final. És fàcil demostrar inductivament que aquestes propietats es cumpleixen per a l'autòmat que descrivim.

	a	b
q_0	q_1	q_0
q_1	q_2	q_0
$\vdots$		
q_i	q_{i+1}	q_0
$\vdots$		
q_{k-1}	q_k	q_0
$\dagger q_k$	q_k	q_0

c) Suposem que existeix un autòmat A amb menys de $k+1$ estats i que reconeix $\{a, b\}^*a^k$. Sigui q_0 l'estat inicial de A. Donat que hi han menys de $k + 1$ estats, dues de les $k + 1$ paraules λ,a,aa,$\ldots$,a^k ens porten al mateix estat. Per tant, existeixen $0 \leqslant i < j \leqslant k$ tals que $q_0a^i = q_0a^j$. Donat que a^k és del llenguatge, q_0a^k és acceptador. Però $q_0a^k = q_0(a^{j+(k-j)}) = q_0(a^ja^{k-j}) = (q_0a^j)a^{k-j} = (q_0a^i)a^{k-j} = q_0(a^ia^{k-j}) = q_0(a^{k-j+i})$. Així doncs, la paraula a^{k-j+i} també ens porta a estat acceptador. Però com que $i < j$, tenim que $k - j + i < k$, de manera que a^{k-j+i} no acaba amb k a's, i per contra és acceptada: contradicció.

Com a resposta alternativa, si apliquem l'algorisme de minimització a l'autòmat de l'apartat anterior, veiem que es comprova que és mínim:

0-distingibles: $\{q_0, \ldots, q_{k-1}\}$ $\{q_k\}$
1-distingibles: $\{q_0, \ldots, q_{k-2}\}$ $\{q_{k-1}\}$ $\{q_k\}$

2-distingibles: $\{q_0, \ldots, q_{k-3}\}\ \{q_{k-2}\}\ \{q_{k-1}\}\ \{q_k\}$

$\vdots$

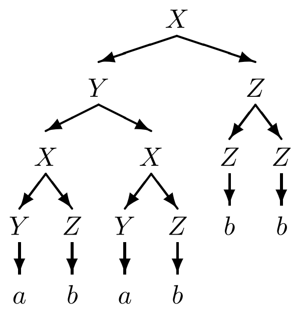
k-distingibles: $\{q_0\}\ \{q_1\} \ldots \{q_{k-1}\}\ \{q_k\}$

d) Ho fem pel lema de bombament. Fixat N, escollim la paraula $w = b^N a^N$, que és de $\{wa^{|w|} | w \in \{a,b\}$, i considerem una factorització qualsevol $w = xyz$ tal que $|xy| \leqslant N$ i $|y| \geqslant 1$. Sota aquestes condicions, x és de la forma b^i per a un cert $i \geqslant 0$, y és de la forma b^j per a un cert $j > 0$, i z és de la forma $b^{N-i-j}a^N$. La paraula xy^2z és $b^{N+j}a^N$, i donat que $j > 0$, també tenim que $N + j > N$, de manera que xy^2z no és de $\{wa^{|w|} | w \in \{a,b\}$. Això conclou la prova.

2. (2 punts) Considerem la següent gramàtica $G_{\texttt{exemple}}$:

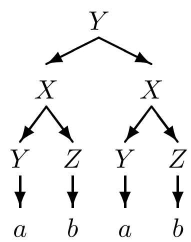
$$\begin{array}{lclcl} X & \to & YZ & & \\ Y & \to & XX & | & a \\ Z & \to & ZZ & | & b \end{array}$$

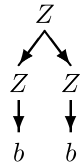
I el següent arbre de derivació del mot terminal $ababbb$ a partir de $G_{\texttt{exemple}}$.



El recorregut en preordre d'aquest arbre és el mot: $XYXYaZbXYaZbZZbZb$.

En general, el recorregut en preordre d'un arbre, es defineix recursivament com el símbol de l'arrel, seguit del recorregut en preordre del primer fill, seguit del recorregut en preordre del segon fill, i així successivament fins al recorregut en preordre de l'últim fill.

Fixeu-vos que per al cas particular del recorregut en preordre de l'arbre de derivació anterior primer ens ve l'arrel X, seguida del recorregut en preordre de  que és $YXYaZbXYaZb$,

seguit del recorregut en preordre de  que és $ZZbZb$.

Donat un arbre de derivació A, anomenem $\texttt{preordre}(A)$ al mot que codifica el recorregut en preordre de A. Donada una gramàtica G, si anomenem $\mathbf{A}(G)$ al conjunt de tots els possibles arbres de derivació de G en mots terminals, llavors $\{\texttt{preordre}(A)|A \in \mathbf{A}(G)\}$ defineix el conjunt dels mots que representen recorreguts en preordre de tots els arbres de derivació de G en mots terminals. Anomenem $\texttt{preordre}(G)$ a aquest conjunt, és a dir, $\texttt{preordre}(G) = \{\texttt{preordre}(A)|A \in \mathbf{A}(G)\}$

Considerant l'exemple de gramàtica anterior $G_{\texttt{exemple}}$, construïu una nova gramàtica que generi $\texttt{preordre}(G_{\texttt{exemple}})$. Fixeu-vos que en la nova gramàtica caldrà que, no només a i b siguin símbols terminals, sinó que també X, Y i Z siguin terminals, quan de fet eren variables de la gramàtica original.

Resposta:

Utilitzarem les variables X', Y' i Z', que generaran $\texttt{Preordre}(L_{G_{\texttt{exemple}},X})$, $\texttt{Preordre}\ (L_{G_{\texttt{exemple}},Y})$ i $\texttt{Preordre}(L_{G_{\texttt{exemple}},X})$, respectivament; on $L_{G,W}$ representa el llenguatge generat per la gramàtica G considerant W com a símbol inicial. La gramàtica que construïm és:

$$\begin{array}{lcl} X' & \to & XY'Z' \\ Y' & \to & YX'X' \mid Ya \\ Z' & \to & ZZ'Z' \mid Zb \end{array}$$

Tot i que es podria demostrar per inducció que es genera $\texttt{preordre}(G_{\texttt{exemple}})$, en aquest cas ens limitem a observar que la utilització d'una regla qualsevol $W \to W_1W_2$ de $G_{\texttt{exemple}}$ porta a un arbre de derivació de la forma

$$\begin{tikzcd} & W \arrow[dl] \arrow[dr] & \\ \alpha_1 & & \alpha_2 \end{tikzcd}$$

on α_1 i α_2 són àrbres de derivació generats desde W_1 i W_2, respectivament, i que el recorregut en preordre d'aquest arbre és $W\texttt{Preordre}(\alpha_1)\texttt{Preordre}(\alpha_2)$. Amb la regla $W' \to WW_1'W_2'$ podem assumir inductivament que W_1' i W_2' generen $\texttt{Preordre}(L_{G_{\texttt{exemple}},W_1'})$ i $\texttt{Preordre}(L_{G_{\texttt{exemple}},W_2'})$.

Com a resposta alternativa podriem haver donat:

$$\begin{array}{lcl} S & \to & XX' \\ X' & \to & YY'ZZ' \\ Y' & \to & XX'XX' \mid a \\ Z' & \to & ZZ'ZZ' \mid b \end{array}$$

3. (1.5 punts) Classifiqueu com a decidible, semi-decidible pero no decidible, o no semi-decidible, els següent problema.

$$\{x|\forall y : (M_x(y)\downarrow\Rightarrow M_x(y) < y)\}$$

(Les entrades i sortides de les màquines de turing s'interpreten com a nombres naturals.)

Resposta:

Demostrarem que no és ni semi-decidible reduïnt desde $\overline{\texttt{K}}$ de la següent manera. Si x és l'element d'entrada, donem com a sortida de la reducció el programa p següent.

```
entrada y
   Simular M_x(x)
   sortida y
```

Si x pertany a $\overline{\mathtt{K}}$, llavors $\mathbf{M}_x(x)\uparrow$, de manera que el programa p no s'atura per cap entrada, i llavors compleix la condició $\forall y : (M_p(y)\downarrow\Rightarrow M_p(y) < y)$, concloent així que p pertany al nostre llenguatge.

En canvi, si x no pertany a $\overline{\mathtt{K}}$, llavors $\mathbf{M}_x(x)\downarrow$, de manera que el programa p s'atura amb totes les entrades, i de fet es comporta com la funció identitat, de manera que no compleix la condició $\forall y : (M_p(y)\downarrow\Rightarrow M_p(y) < y)$, concloent així que p no pertany al nostre llenguatge.

Com a resposta alternativa, pel teorema de Rice, donat que L és un conjunt d'índexos, que no és ni el total ni el buit, i que conté els nombres que representen la funció totalment indefinida, resulta que L no és ni semi-decidible.

Més en detall, L és un conjunt d'indexos perque en la condició sempre que apareix x, ho fa amb el contexte M_x, és a dir, una condició només sobre el comportament del programa M_x, independentment de la seva implementació. L no és buit perque conté, per exemple, a qualsevol nombre de programa que es penja sempre. L no és el total perque no conté per exemple als nombres de programes que donen com a sortida la pròpia entrada, per a totes les entrades.

4. (2 punts en total) **En aquest cas heu d'escollir qué contestar d'entre dos problemes alternatius**

Opció 1:

En els exercicis-control del curs hem vist que és indecidible el problema de, donades dues gramàtiques, saber si tenen intersecció no buida.

$$\{\langle G_1, G_2\rangle \mid \mathbf{L}(G_1)\cap\mathbf{L}(G_2)\neq\emptyset\}$$

Ara volem demostrar que també ho és el problema de, donades dues gramàtiques, saber si tenen com a intersecció un nombre infinit de mots; és a dir, si hi ha infinits mots generats per totes dues gramàtiques.

$$\{\langle G_1, G_2\rangle \mid |\mathbf{L}(G_1)\cap\mathbf{L}(G_2)| = \infty\}$$

a) (1) Demostreu amb un contraexemple que el següent intent de reducció desde intersecció no buida no funciona. (És a dir, que no és cert que una entrada del problema del intersecció no buida té resposta afirmativa si i només si la seva transformada en una entrada del problema que hem enunciat també en té).

   Donada la entrada $\langle G_1, G_2\rangle$ de intersecció no buida, construïm una nova entrada $\langle G_1', G_2'\rangle$ per a intersecció infinita així:

   Si S_1 és el símbol inicial de G_1, llavors G_1' té una nova variable S_1' com a símbol inicial, i les seves regles són les de G_1 més $S_1'\to S_1S_1'|\lambda$.

   Si S_2 és el símbol inicial de G_2, llavors G_2' té una nova variable S_2' com a símbol inicial, i les seves regles són les de G_2 més $S_2'\to S_2S_2'|\lambda$.

b) (1) Doneu una reducció des del problema de intersecció no buida i justifiqueu que és correcta. Pista: és recomanable utilitzar algún símbol nou per a aconseguir-ho.

Opció 2:

Expliqueu formalment, amb generalitat i claretat, cóm es pot obtenir un autòmat amb pila indeterminista que reconegui el llenguatge generat per una gramàtica donada.

Resposta:

Opció 1:

a) **Podem tenir com a entrada les gramàtiques $S_1 \to a$ i $S_2 \to aa$. La intersecció és buida, i en canvi les corresponents**
$$\begin{array}{lcl} S_1' & \to & S_1S_1' \mid \lambda \\ S_1 & \to & a \end{array}$$
i
$$\begin{array}{lcl} S_2' & \to & S_2S_2' \mid \lambda \\ S_2 & \to & aa \end{array}$$
Generen infinites paraules comunes: $\{a^{2n}\}$.

b) **Donada la entrada $\langle G_1, G_2\rangle$ de intersecció no buida, agafem un símbol nou # que no aparegui ni a G_1 ni a G_2, i construïm una nova entrada $\langle G_1', G_2'\rangle$ per a intersecció infinita així:**
Si S_1 és el símbol inicial de G_1, llavors G_1' té una nova variable S_1' com a símbol inicial, i les seves regles són les de G_1 més $S_1' \to S_1\#S_1'|S_1$.
Si S_2 és el símbol inicial de G_2, llavors G_2' té una nova variable S_2' com a símbol inicial, i les seves regles són les de G_2 més $S_2' \to S_2\#S_2'|S_2$.
Si G_1 i G_2 generen una paraula comuna w, llavors G_1' i G_2' generen infinites paraules comunes de la forma $w\#w\#\ldots\#w\#w$.
Per la direcció contraria, si G_1' i G_2' generen infinites paraules comunes, agafem qualsevol d'elles, que serà de la forma w o $w\#\ldots$, on w no conté $\#$. Aquesta w és generable tant desde G_1' com desde G_2'.

Com a resposta alternativa també podiem haver creat com a G_1' les regles de G_1 més $S_1' \to \#S_1'|S_1$, i com a G_2' les regles de G_2 més $S_2' \to \#S_2'|S_2$.

Opció 2:

A partir d'una CFG $G = \langle V, \Sigma, P, S\rangle$ podem construir de la manera següent un NPDA $M = \langle Q, \Sigma, \Gamma, \delta, q_0, Z_0, F\rangle$ tal que $L(M) = L(G)$:

- **el conjunt d'estats $Q = \{q_0, p, f\}$ (estat inicial, estat de procés i estat acceptador);**
- **l'alfabet de pila $\Gamma = \Sigma \cup V \cup \{Z_0\}$ on $Z_0 \notin \Sigma \cup V$ és el símbol de fons de pila;**
- **el conjunt d'estats acceptadors $F = \{f\}$;**
- **la funció de transició δ està format per:**
  $Z_0q_0 \vdash Z_0Sp$ **(una λ-transició que posa la variable inicial al cim de la pila en l'estat de procés);**
  $\{Zp \vdash \alpha^R p|\ Z \to \alpha \in P\}$ **(una λ-transició per a cada producció de la variable Z al cim de la pila);**
  $\{apa \vdash p|\ a \in \Sigma\}$ **(si el cim de la pila és un terminal, l'autòmat ha de verificar que sigui igual al símbol en curs del mot d'entrada i avançar el capçal);**
  $Z_0p \vdash f$ **(una λ-transició que permet passar a l'estat acceptador quan tot el mot ha estat processat).**

(Examen de Gener-2008)

Tingueu present en tots els exercicis que les respostes s'han de justificar. La mera presentació d'una proposta de resultat no és suficient.

1. (2.5 punts en total) Atenció, el segon punt d'aquest problema s'ha de contestar i entregar en el propi enunciat.

   - (1) Considerem el llenguatge sobre $\{a, b\}^*$ dels mots que acaben en *abaab*. Definiu formalment aquest conjunt amb la notació clàssica de conjunts, i trobeu l'autòmat determinista mínim que el genera, directament, o per transformació de la seva formalització, aplicant després operacions de tancament i minimitzant.
   - (1.5) Donat un mot w sobre $\{a, b\}^*$, el llenguatge dels mots que acaben en w, és a dir $\{a, b\}^*w$, és regular. De fet, es pot demostrar que existeix un autòmat determinista amb $|w| + 1$ estats que el reconeix. Però ara ens ocuparem de veure que no se'n pot trobar cap de més petit. A continuació hi ha una demostració (amb algunes parts esborrades) que per a qualsevol $w \in \{a, b\}^*$, no hi ha cap autòmat determinista amb menys que $|w| + 1$ estats que reconeix $\{a, b\}^*w$. Afegiu el que hi falta.

     Sigui $n = |w|$. Suposem que hi ha un DFA A amb menys de $n + 1$ estats que reconeix el llenguatge. Sigui q_0 l'estat inicial de A. Com que hi ha $n + 1$ mots a `prefixos`(w), dos d'ells ens porten al mateix estat; els anomenem u i v. Per tant, $q_0u = q_0v$. Sense pèrdua de generalitat assumim que u és prefix propi de v (en altre cas v seria prefix propi de u i es procediria anàlogament). Així doncs, v és de la forma uu', per $|u'|$..........0, i w és de la forma $vv' = uu'v'$ per $|v'|$..........0. Com que w acaba en ...., tenim que (q_0w) és acceptador. Però com que $q_0w = q_0(vv') = (q_0v)v' = (..........)v'$, a aquest mateix estat acceptador també s'hi arriba amb .... Donat que $|u'|$..........0 tenim que $|uv'| < |u..........v'| = |w|$. Per tant, uv' no pot acabar en .... perquè és més petit que ...., i per contra s'està acceptant: contradicció.

Resposta:

- **(versió simplificada) Definim el conjunt així: $\{w \in \{a, b\}^* \mid \exists x \in \{a, b\}^* : (w = xabaab)\}$. Aquest llenguatge es pot escriure com la següent concatenació: $\{a, b\}^*\{abaab\}$. Després d'escriure els corresponents autòmats, calcular l'autòmat concatenació i minimitzar, obtenim:**

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

  Si ja hem obtingut aquest mateix autòmat bans de minimitzar, en realitat no cal fer-ho, doncs té $|w| + 1 = 6$ estats, i en virtud de l'apartat següent, és mínim.

- **Sigui $n = |w|$. Suposem que hi ha un DFA A amb menys de $n + 1$ estats que reconeix el llenguatge. Sigui q_0 l'estat inicial de A. Com que hi ha $n + 1$ mots a `prefixos`(w), dos d'ells ens porten al mateix estat, que anomenem u i v. Per tant, $q_0u = q_0v$. Sense pèrdua de generalitat assumim que u és prefix propi de v (en altre cas v seria prefix propi de u i es procediria anàlogament). Per tant, v és de la forma uu', per $|u'| > 0$, i w és de la forma $vv' = uu'v'$ per $|v'| \geqslant 0$. Com que w acaba en w, tenim**

que (q_0w) és acceptador. Però com que $q_0w = q_0(vv') = (q_0v)v' = (q_0u)v'$, a aquest mateix estat acceptador també s'hi arriba amb uv'. Donat que $|u'| > 0$ tenim que $|uv'| < |uu'v'| = |w|$. Per tant, uv' no pot acabar en w perquè és més petit que w, i per contra s'està acceptant: contradicció.

2. (2 punts en total) Definim la mida d'una gramàtica G, que denotem $|G|$, com el nombre total de símbols que apareixen a les parts dretes de les regles de G. Per exemple, la gramàtica $S \to (S)S|\lambda$ té mida 4 perquè $|(S)S| + |\lambda| = 4 + 0 = 4$.

   - (1 punt) Demostreu que per a qualsevol gramàtica G existeix una gramàtica G' tal que $\mathtt{L}(G) = \mathtt{L}(G')$, totes les parts dretes de G' tenen mida menor o igual a 2 (més formalment: $\forall X \to \alpha \in \mathtt{regles}(G') : |\alpha| \leqslant 2$), i $|G'| \leqslant 2|G|$.
   - (1 punt) Demostreu que, per a qualsevol gramàtica G que compleixi $\lambda \notin \mathtt{L}(G)$, existeix una gramàtica G' sense λ-produccions (és a dir, les de la forma $X \to \lambda$) tal que $\mathtt{L}(G') = \mathtt{L}(G)$ i $|G'| \leqslant 4|G|$.

   Resposta:

   - **Construïm G' a partir de G de la següent manera. Per cada regla $X \to \alpha_1 \dots \alpha_n$ de G amb $n \geqslant 3$, on els α_i són els terminals i no terminals concatenats que apareixen a la part dreta de la regla, ens inventem (n-2) variables noves $X_2, \dots, X_{n-1}$, esborrem la regla anterior, i afegim les regles $X \to \alpha_1 X_2,\ X_2 \to \alpha_2 X_3,\ X_3 \to \alpha_3 X_4,\ \dots,\ X_{n-1} \to \alpha_{n-1}\alpha_n$. És obvi que desde $\alpha_1 \dots \alpha_n$ i $\alpha_1 X_2$ es genera el mateix llenguatge. Per tant, la nostra transformació de G en G' preserva el llenguatge. Ademés, la mida de la regla original era n, i la suma de mides de les corresponents noves regles és $2 * (n-1) < 2n$. Així doncs, $|G'| \leqslant 2|G|$.**
   - **Donada G, obtenim primer G'' segons la transformació de l'apartat anterior. Així doncs, $\mathtt{L}(G) = \mathtt{L}(G'')$, $|G''| \leqslant 2|G|$, i totes les parts dretes de G'' tenen com a molt mida 2. Un cop fet això, apliquem la transformació clàssica per a eliminar λ-produccions. En el nostre cas particular, això correspon a fer el següent. Per cada regla de la forma $X \to \alpha_1\alpha_2$, si α_1 és anulable, llavors hem d'afegir la regla $X \to \alpha_2$, i si α_2 és anulable, llavors hem d'afegir la regla $X \to \alpha_1$. Finalment, hem d'esborrar totes les produccions de la forma $Z \to \lambda$. Aquesta transformació de G'' en una nova gramàtica G' preserva el llenguatge, com ja és sabut. Ademés, com a molt, per cada regla de mida 2 s'afegeixen dues regles de mida 1. Per tant, $|G'| \leqslant 2|G''| \leqslant 4|G|$.**

3. (1.5 punts) Demostreu que el llenguatge L següent no és regular.

$$L = \{a^x b^y c^z d^t \mid x, y, z, t \geqslant 0 \ \wedge\ ((x = t) \Rightarrow (y = z))\} \cup a^*b^*c^* \cup b^*c^*d^*$$

   Resposta: Suposem que L és regular. Intersequem L amb ab^*c^*d, que és regular, obtenint així $L' = \{ab^nc^nd\}$. Donat que la intersecció de llenguatges regulars dona lloc a un llenguatge regular, resulta que L' és regular. Apliquem el morfisme definit per $\sigma(a) = \lambda$, $\sigma(b) = a$, $\sigma(c) = b$, $\sigma(d) = \lambda$ sobre L' obtenint així $\{a^nb^n\}$. Donat que l'aplicació d'un morfisme sobre un llenguatge regular dona lloc a un llenguatge regular, resulta que $\{a^nb^n\}$ és regular. Però $\{a^nb^n\}$ es un conegut llenguatge no regular, contradicció.

   Resposta alternativa 1: Suposem que L és regular. Llavors, existeix un DFA A que el reconeix. Sigui N el nombre d'estats de A. Donat que la paraula ab^Nc^Nd és de L, llavors és acceptada per A. En la execució de A amb entrada ab^Nc^Nd, passem almenys per aquests estats: $q_0a, q_0ab, q_0abb, \dots, q_0ab^N$. En total en són $N{+}1$ estats, i per tant n'hi ha un de repetit. Així doncs, existeixen i, j cumplint $0 \leqslant i < j \leqslant N$ tals que $q_0ab^i = q_0ab^j$. Donat que $q_0ab^Nc^Nd$

és acceptador, també ho és $q_0ab^Nc^Nd = (q_0ab^i)b^{N-i}c^Nd = (q_0ab^j)b^{N-i}c^Nd = q_0(ab^{N-i+j}c^Nd)$. Així doncs, $ab^{N-i+j}c^Nd$ ens porta a estat acceptador, però donat que $j > i$, aquesta paraula no és del llenguatge, contradicció.

Resposta alternativa 2: Apliquem el lema de bombament. Fixat N, agafem la paraula $w = a^Nbccd^{N+N!}$, que és del llenguatge. Considerem una factorització qualsevol $w = xyz$ cumplint $|xy| \leqslant N$ i $|y| \geqslant 1$. Necessàriament, existeixen j, k cumplint $x = a^j$, $y = a^k$, $k > 1$, $j + k \leqslant N$. Busquem un i tal que $(i-1)k = N!$. Aquest és $i = 1 + (N!)/k$, que és un nombre natural perquè $N!$ és un producte que conté k entre els seus factors. La paraula $xy^iz = a^{N+N!}bccd^{N+N!}$ no és del llenguatge, i això conclou la prova.

4. (1.5 punts) Classifiqueu com a decidible, semi-decidible però no decidible, o no semi-decidible, el problema següent.

$$C = \{\langle x, y\rangle \mid \forall z : (M_x(z)\downarrow \Leftrightarrow M_y(z)\downarrow)\}$$

Resposta: Demostrem que no és ni semi-decidible reduïnt desde $\overline{\texttt{K}} = \{x | M_x(x)\uparrow\}$. La reducció consisteix en generar, per a cada x, la parella $\langle p(x), q\rangle$, on q és un nombre que codifica un programa que no s'atura per a cap entrada, i $p(x)$ és el programa següent:

```
entrada y
   Simular Mx(x)
   sortida y
```

Si un cert x pertany a $\overline{\texttt{K}}$, llavors $M_{p(x)}$ no s'atura per a cap entrada, i per tant $\langle p(x), q\rangle$ pertany a C. Si x no pertany a $\overline{\texttt{K}}$, llavors $M_{p(x)}$ és una màquina que s'atura per a totes les entrades, i per tant $\langle p(x), q\rangle$ no pertany a C. Això conclou la prova.

5. (2.5 punts en total) Durant el curs hem vist que és indecidible el problema de, donades dues gramàtiques, saber si tenen intersecció no buida.

$$\{\langle G_1, G_2\rangle \mid \texttt{L}(G_1) \cap \texttt{L}(G_2) \neq \emptyset\}$$

- (1.5) Demostreu que també ho és el problema de, donades dues gramàtiques, saber si tenen com a intersecció algun mot de longitud parell. Feu una reducció des del problema d'intersecció no buida. (anomenem *intersecció parella* a aquest nou problema.)

$$\{\langle G_1, G_2\rangle \mid \exists w : (|w| \in \dot{2} \wedge w \in (\texttt{L}(G_1) \cap \texttt{L}(G_2)))\}$$

- (1) Demostreu que també és indecidible el problema de, donades dues gramàtiques, saber si tenen com a intersecció dos o més mots de longitud parell. (anomenem *2-intersecció parella* a aquest nou problema.)

$$\{\langle G_1, G_2\rangle \mid \exists w_1, w_2 : (w_1 \neq w_2 \wedge |w_1|, |w_2| \in \dot{2} \wedge w_1, w_2 \in (\texttt{L}(G_1) \cap \texttt{L}(G_2)))\}$$

Resposta:

- **Donada la entrada $\langle G_1, G_2\rangle$ d'intersecció no buida, construïm una nova entrada $\langle G_1', G_2'\rangle$ per a intersecció parella com segueix. Ens inventem un símbol nou $\#$. Si S_1 és el símbol inicial de G_1, llavors G_1' té una nova variable S_1' com a símbol inicial, i les seves regles són les de G_1 més $S_1' \to S_1\#\#S_1$. Si S_2 és el símbol inicial de G_2, llavors**

G_2' té una nova variable S_2' com a símbol inicial, i les seves regles són les de G_2 més $S_2' \to S_2\#\#S_2$.
Si G_1 i G_2 generen una paraula comuna w, llavors G_1' i G_2' generen la paraula comuna $w\#\#w$ de longitud parella. Per a la direcció contrària, si G_1' i G_2' generen una paraula comuna u de longitud parella, llavors, per la forma d'aquestes gramàtiques, u és de la forma $u_1\#\#u_2$, on tant u_1 com u_2 són generables tant amb G_1 com amb G_2. Per tant, G_1 i G_2 generen alguna paraula comuna, i això conclou la prova.

- **Reduïm desde el llenguatge de l'apartat anterior. Donada la entrada $\langle G_1', G_2' \rangle$ d'intersecció parella, construïm una nova entrada $\langle G_1'', G_2'' \rangle$ per a 2-intersecció parella com segueix. Ens inventem un símbol nou \$. Si S_1' és el símbol inicial de G_1', llavors G_1'' té una nova variable S_1'' com a símbol inicial, i les seves regles són les de G_1' més $S_1'' \to S_1'|\$\$$. Si S_2' és el símbol inicial de G_2', llavors G_2'' té una nova variable S_2'' com a símbol inicial, i les seves regles són les de G_2' més $S_2'' \to S_2'|\$\$$.**
Si G_1' i G_2' generen una paraula comuna de longitud parella, llavors G_1'' i G_2'' també la generen, i ademés totes dues generen també \$\$, que és de longitud parella. Per a la direcció contrària, si G_1'' i G_2'' generen dues paraules comunes de longitud parella, llavors almenys una no és \$\$, i ademés és generable desde G_1' i G_2'. Això conclou la prova.

(Examen de Juny-2008)

1. (2 punts en total, temps màxim de resolució estimat: 40 minuts) Contesteu amb una, dues o tres frases, i sobre el propi full de l'enunciat, les qüestions següents. (que no requereixen justificació, tret que la pregunta indiqui el contrari).

   (a) (0.25) Doneu un exemple de llenguatge que sigui decidible però no incontextual.

   Resposta: $\{a^n b^n c^n \mid n \geqslant 0\}$.

   (b) (0.25) Doneu un exemple de llenguatge que sigui incontextual i inherentment ambigu.

   Resposta: $\{a^n b^n c^m \mid n, m \geqslant 0\} \cup \{a^m b^n c^n \mid n, m \geqslant 0\}$

   (c) (0.5) Acabeu la següent demostració del fet que $\{a^n b^n | n \geqslant 0\}$ és no-regular: **suposem que $\{a^n b^n | n \geqslant 0\}$ és regular. Llavors existeix un autòmat A que el reconeix. Sigui N el nombre d'estats de A, i q_0 l'estat inicial de A. L'autòmat A, amb entrada $a^N b^N$ passa almenys pels estats $q_0, q_0 a, q_0 aa, \ldots, q_0 a^N$. Com que n'hi ha $N+1$, algun està repetit. . . .**

   Resposta: Per tant existeixen $0 \leqslant i < j \leqslant N$ tals que $q_0 a^i = q_0 a^j$. Així doncs, $q_0(a^i b^i) = (q_0 a^i) b^i = (q_0 a^j) b^i = q_0(a^j b^i)$. Però llavors aquest estat hauria de ser acceptador i rebutjador al mateix temps, contradicció.

   (d) (0.25) Donats dos autòmats $A_1 = \langle \Sigma, Q_1, \delta_1, q_{i1}, F_1 \rangle$ i $A_2 = \langle \Sigma, Q_2, \delta_2, q_{i2}, F_2 \rangle$, reconeixent llenguatges L_1 i L_2, un autòmat que reconeix $L = L_1 \cap L_2$ s'obté definint com a estats totes les parelles d'estats $Q = Q_1 \times Q_2$, definint les transicions com $\delta(\langle q_1, q_2 \rangle, a) = \langle \delta_1(q_1, a), \delta_2(q_2, a) \rangle$, i escollint com a estats inicial i acceptadors . . .

   Resposta: $\langle q_{i1}, q_{i2} \rangle$ com a inicial i $F_1 \times F_2$ com a acceptadors.

(e) (0.25) Si el llenguatge L és semi-decidible i el seu complementari $\overline{L}$ també, què en podem deduir?

Resposta: Que L és decidible.

(f) (0.5) Suposem que A és un DFA mínim que reconeix un cert llenguatge L. El DFA A' obtingut intercanviant estats acceptadors per no acceptadors de A reconeix el complementari $\overline{L}$. A continuació es comença a justificar que A' també resulta ser un DFA mínim. Acabeu la justificació. **Suposem que A' no és el DFA mínim reconeixent $\overline{L}$. Llavors, existeix un DFA més petit A'' que també reconeix $\overline{L}$. Si intercambiem estats acceptadors per no acceptadors en A''...**

Resposta: Obtenim un nou DFA A''' que accepta L i amb menys estats que A, cosa que contradiu el fet que A fos mínim.

2. (3.5 punts en total, temps màxim de resolució estimat: 60 minuts) Diem que un llenguatge és un *singletó* si conté exàctament un mot, i.e. és de la forma $\{w\}$. A continuació definim el concepte de gramàtica singletó, que és una gramàtica pensada per tal que el seu llenguatge generat sigui singletó.

Una gramàtica singletó té els símbols no terminals (variables) indexats de la forma $X_1, \ldots, X_n$. Cada variable X_i apareix com a part esquerra de regla en exàctament una regla, que pot ser de la forma $X_i \to c$ per a un cert símbol terminal c, o de la forma $X_i \to X_j X_k$ per a $j, k < i$. El símbol inicial de la gramàtica és X_n.

Per exemple, la següent gramàtica singletó genera $\{aabaab\}$.

$$\begin{array}{lcl} X_5 & \to & X_4X_4 \\ X_4 & \to & X_2X_3 \\ X_3 & \to & b \\ X_2 & \to & X_1X_1 \\ X_1 & \to & a \end{array}$$

Donada una gramàtica singletó G i una de les seves variables X_i, denotem amb $\texttt{mot}(G, X_i)$ al mot generat des de X_i. Observeu que si G conté una regla de la forma $X_i \to c$, llavors $\texttt{mot}(G, X_i) = c$, i que si G conté una regla de la forma $X_i \to X_j X_k$, llavors $\texttt{mot}(G, X_i) = \texttt{mot}(G, X_j)\texttt{mot}(G, X_k)$.

La gràcia de les gramàtiques singletó és que permeten descriure mots "llargs" utilitzant una representació "curta", i a més, permeten calcular ràpidament propietats sobre el mot sense haver de generar-lo (cosa que seria molt costosa si el mot és "llarg").

(a) (0.5) Escriviu una gramàtica singletó per a *ababab* utilitzant només 5 variables.

(b) (1) Descriviu una gramàtica singletó per a $a^{(2^n)}$ utilitzant només $n + 1$ variables.

(c) (1) L'exemple de l'apartat anterior mostra que les gramàtiques singletó poden comprimir molt. Però aquesta compressió té un límit. Volem justificar que una gramàtica singletó amb n variables $X_1, \ldots, X_n$ pot arribar a definir un mot de com a molt mida 2^{n-1}. Feu-ho per inducció sobre la definició de la gramàtica i la forma de les regles, veient que, per tot i entre 1 i n, tenim que $|\texttt{mot}(G, X_i)| \leqslant 2^{i-1}$.

(d) (1) Ja hem vist que podem comprimir molt, però fins a un cert límit. Ara veiem que podem fer un càlcul sobre el mot generat, però sense necessitat de generar-lo. Donat un autòmat A i una gramàtica singletó G amb variables $X_1, \ldots, X_n$, volem comprobar si A accepta $\texttt{mot}(G, X_n)$, però sense generar-lo, ja que podria tenir mida 2^n i això implicaria cost exponencial. Doneu un algorisme de cost polinòmic que calculi a quin estat arriba A amb entrada $\texttt{mot}(G, X_n)$. (Pista: necessitareu calcular certa informació per cada $\texttt{mot}(G, X_i)$. Utilitzeu la definició de la gramàtica i la forma de les regles.)

Resposta:

(a)

$$\begin{array}{lcl} X_5 & \to & X_3X_4 \\ X_4 & \to & X_3X_3 \\ X_3 & \to & X_1X_2 \\ X_2 & \to & b \\ X_1 & \to & a \end{array}$$

(b)

$$\begin{array}{lcl} X_{n+1} & \to & X_nX_n \\ X_n & \to & X_{n-1}X_{n-1} \\ \vdots & & \vdots \\ X_2 & \to & X_1X_1 \\ X_1 & \to & a \end{array}$$

(c) **Demostrem per inducció sobre i que cada X_i cumpleix $|\texttt{mot}(G, X_i)| \leqslant 2^{i-1}$.**

Sigui un i qualsevol entre 1 i n. Assumim inductivament que $|\texttt{mot}(G, X_j)| \leqslant 2^{j-1}$ per a qualsevol $1 \leqslant j < i$. Distingim dos casos. Si la regla de X_i és de la forma $X_i \to c$, llavors $\texttt{mot}(G, X_i) = c$, i per tant $|\texttt{mot}(G, X_i)| = 1 \leqslant 2^{i-1}$. En canvi, si la regla de X_i és de la forma $X_i \to X_j X_k$, llavors $\texttt{mot}(G, X_i) = \texttt{mot}(G, X_j)\texttt{mot}(G, X_k)$ i per tant $|\texttt{mot}(G, X_i)| = |\texttt{mot}(G, X_j)| + |\texttt{mot}(G, X_k)| \leqslant 2^{j-1} + 2^{k-1} \leqslant 2^{i-2} + 2^{i-2} = 2^{i-1}$.

(d) **Sigui δ la funció de transició de A. El que farem és calcular $\delta(q, \texttt{mot}(X_i))$ per a tots els estats q de A i tots els X_i. Si som capaços de fer això ja tindrem la resposta, doncs n'hi haurà prou amb comprobar que $\delta(q_0, \texttt{mot}(X_i))$ és acceptador, on q_0 és l'estat inicial de A. Aquest càlcul el fem per ordre, desde i igual a 1 fins a n. Per a cada X_i, assumim ja calculats els $\delta(q, \texttt{mot}(X_j))$ per als $j < i$ i tots els estats q. Així doncs, n'hi ha prou amb veure que cada $\delta(q, \texttt{mot}(X_i))$ es pot calcular eficientment sota aquestes assumcions. Distingim dos casos. Si la regla de X_i és de la forma $X_i \to c$, llavors $\texttt{mot}(G, X_i) = c$, i per tant $\delta(q, \texttt{mot}(X_i)) = \delta(q, c)$. Donat que $\delta(q, c)$ s'obté diréctament de l'autòmat d'entrada, aquest càlcul es pot fer eficientment. En canvi, si la regla de X_i és de la forma $X_i \to X_j X_k$, llavors $\texttt{mot}(G, X_i) = \texttt{mot}(G, X_j)\texttt{mot}(G, X_k)$, i per tant $\delta(q, \texttt{mot}(X_i)) = \delta(q, \texttt{mot}(G, X_j)\texttt{mot}(G, X_k)) = \delta(\delta(q, \texttt{mot}(G, X_j)), \texttt{mot}(G, X_k))$. Sigui q' l'estat $\delta(q, \texttt{mot}(G, X_j))$. Per les nostres hipòtesis, q' ja està precalculat, i també $q'' = \delta(q', \texttt{mot}(G, X_k))$, que és el que voliem obtenir.**

3. (2 punts, temps màxim de resolució estimat: 30 minuts) Classifiqueu com a decidible, semi-decidible però no decidible, o no semi-decidible, el problema següent: $C = \{\langle x, y\rangle \mid \exists z : (M_x(z) \downarrow \ \wedge M_y(z) \downarrow \ \wedge M_x(z) < M_y(z))\}$

Resposta:

Demostrem que és indecidible reduïnt desde $\texttt{K}$. La reducció consisteix a, per cada x d'entrada, generar el parell $\langle p(x), q\rangle$, on $p(x)$ és número que codifica el programa

```
entrada y
   Simular Mx(x)
   sortida 1
```

i q és el número que codifica el programa

```
entrada y
   sortida 2
```

Si x pertany a $\texttt{K}$, llavors $\texttt{M}_x(x) \downarrow$, i per tant el programa codificat per $p(x)$ dona sempre sortida 1. Com que el programa codificat per q dona sempre sortida 2, resulta que el parell $\langle p(x), q\rangle$ és del llenguatge C.

Si x no pertany a $\texttt{K}$, llavors $\texttt{M}_x(x) \uparrow$, i per tant el programa codificat per $p(x)$ no s'atura per a cap entrada. Així doncs, el parell $\langle p(x), q\rangle$ no és del llenguatge C.

Ara justificarem que el llenguatge anterior és semi-decidible. N'hi ha prou amb mostrar un programa que semi-decideix el conjunt.

```
entrada ⟨x, y⟩
   t:=1
   Executar indefinidament:
      Per a cada i entre 0 i t fer:
         Si M_x(i) i M_y(i) s'aturen en t passos i M_x(i) < M_y(i) llavors acceptar
      t:=t + 1
```

4. (2.5 punts en total, temps màxim de resolució estimat: 50 minuts) Durant el curs s'ha vist que els següents dos problemes són indecidibles:

   - (intersecció no buida) Donades dues gramàtiques, saber si tenen intersecció no buida.

   $$\{\langle G_1, G_2\rangle \mid \mathtt{L}(G_1) \cap \mathtt{L}(G_2) \neq \emptyset\}$$

   - (accessibilitat de mots) Donada una llista finita R de regles de reescriptura (un conjunt finit de parells de mots (u_i, v_i)) i dos mots u (inicial), v (final), determinar si és possible passar de u a v utilitzant les regles de reescriptura (un seguit de substitucions d'algun submot u_i pel seu corresponent v_i).

   $$\{\langle R, u, v\rangle \mid u \to_R^* v\}.$$

   Volem donar reduccions entre aquests dos problemes.

   (a) Primer volem veure que es pot reduïr fàcilment el primer problema (intersecció no buida) al segon (accessibilitat de mots). Per a fer-ho, us fem una proposta de com començar la reducció, i hi heu d'afegir el que hi falta.

   Reducció: Sigui $\langle G_1, G_2\rangle$ el parell de gramàtiques que són instància d'intersecció no buida. Per simplificar, suposarem que els noms de variables que apareixen a G_1 són disjunts dels noms de variables que apareixen a G_2 (si calgués, es poden reanomenar). El nostre objectiu és generar una entrada d'accessibilitat de mots adequada.

   Sigui S_1 el símbol inicial de G_1, i S_2 el símbol inicial de G_2. El que farem és generar una entrada de la forma $\langle R, S_1, S_2\rangle$, és a dir, un sistema de reescriptura R, i el parell de mots S_1 i S_2. Queda per determinar R per tal que la reducció sigui correcta, i això és el que heu de fer. És a dir, descriviu com s'obté R a partir de les regles de G_1 i G_2, i justifiqueu que $S_1 \to_R^* S_2$ si i només si G_1 i G_2 tenen intersecció no buida.

   (descriure R encertadament val 1 punt, i la justificació val 0.5 punts.)

   Resposta: Definim la llista de regles R com la llista de regles/produccions de G_1, més la llista de regles/produccions de G_2 invertides. Formalment, si R_1 són les produccions de G_1 i R_2 són les produccions de R_2, llavors definim $R = \{X \to \alpha | X \to \alpha \in R_1\} \cup \{\alpha \to X | X \to \alpha \in R_2\}$. En altres paraules, $R = R_1 \cup R_2^{-1}$ amb una definició natural d'invers adequada.

   Per a veure que la reducció és correcta, mirem les dues direccions. Suposem que G_1 i G_2 tenen intersecció no buida. Llavors existeix un mot comú w generat per totes dues gramàtiques. Per tant, $S_1 \to_{R_1}^* w$ i $S_2 \to_{R_2}^* w$. Això implica $S_1 \to_{R_1}^* w \to_{R_2^{-1}}^* S_2$, de manera que concloem $S_1 \to_R^* S_2$. Per a la direcció contrària, suposem $S_1 \to_R^* S_2$. El conjunt R consta de les regles de R_1 i de les regles de R_2^{-1}. El primer que fem és observar que una aplicació d'una regla de R_2^{-1} seguida d'una aplicació d'una regla de R_1 es poden commutar. És a dir, que si tenim $u \to_{R_2^{-1}} v \to_{R_1} w$, llavors existeix v' tal que $u \to_{R_1} v' \to_{R_2^{-1}} w$. Per a veure-ho, siguin $\alpha \to X$ i $Y \to \beta$ les corresponents regles de R_2^{-1} i R_1 aplicades. Llavors la derivació anterior és de la forma o bé $u = u_1\alpha u_2 Y u_3 \to_{R_2^{-1}}$

$u_1Xu_2Yu_3 \rightarrow_{R_1} u_1Xu_2\beta u_3 = w$, **o bé** $u = u_1Yu_2\alpha u_3 \rightarrow_{R_2^{-1}} u_1Yu_2Xu_3 \rightarrow_{R_1} u_1\beta u_2Xu_3 = w$. **Podem construir la derivació que volem, així: o bé** $u = u_1\alpha u_2Yu_3 \rightarrow_{R_1} u_1\alpha u_2\beta u_3 \rightarrow_{R_2^{-1}} u_1Xu_2\beta u_3 = w$, **o bé** $u = u_1Yu_2\alpha u_3 \rightarrow_{R_1} u_1\beta u_2\alpha u_3 \rightarrow_{R_2^{-1}} u_1\beta u_2Xu_3 = w$. **Un cop que hem vist que podem commutar els passos de reescriptura, concloem que, a partir de la derivació** $S_1 \rightarrow_R^* S_2$ **en podem construir una de la forma** $S_1 \rightarrow_{R_1}^* w \rightarrow_{R_2^{-1}}^* S_2$. **La paraula** w **no pot tenir variables de** G_1 **perque** G_1 **i** G_2 **no comparteixen variables, i per tant les regles de** R_2^{-1} **no les podrien eliminar, de manera que no podriem arribar a** S_2. **Anàlogament es pot veure que** w **no pot contenir variables de** G_2. **Per tant,** w **és un mot terminal arribable desde** S_1 **amb** R_1 **i desde** S_2 **amb** R_2. **Això implica que** G_1 **i** G_2 **generen el mot comú** w, **i això conclou la prova.**

(b) Ara volem donar una reducció del segon problema (accessibilitat de mots) al primer (intersecció no buida). Això és el que es fa a continuació.

Sigui $\langle R = \langle u_1 \rightarrow v_1, \ldots, u_n \rightarrow v_n\rangle, u, v\rangle$ **una entrada d'accessibilitat de mots. Per simplificar suposem que** R **conté la regla** $u \rightarrow u$. **Si no hi és, la podem afegir sense problemes, ja que no modifica que** v **sigui accessible des de** u. **Gràcies a aquesta regla comodí podrem afirmar tranquilament que, si** v **és accessible des de** u, **llavors existeix una derivació de** u **a** v **amb almenys un pas de reescriptura.**

La idea de la reducció que intentarem és la següent. A partir de R, u **i** v, **obtindrem dues gramàtiques** G_1, G_2 **tals que, donada una derivació per reescriptura** $u = s_1 \rightarrow_R s_2 \rightarrow_R \ldots \rightarrow s_m = v$ **amb** $m \geqslant 2$, G_1 **i** G_2 **podran generar el mot comú** $s_1\#s_2\#s_3\#\ldots s_m\#s_m^r\ldots\#s_3^r\#s_2^r\#s_1^r$, **on # és un símbol nou, i l'exponent** r **significa el revessat. De fet,** G_1 **generarà mots qualssevols de la forma** $w_1\#w_2\#w_3\#\ldots\#w_k\#w_k^r\#\ldots\#w_3^r\#w_2^r\#w_1^r$ **amb** $k \geqslant 2$ **complint que** w_1 **és** u, **i** w_k **és** v. **Per altra banda,** G_2 **generarà mots qualssevols de la forma** $w_1\#w_2\#w_3\#\ldots\#w_k\#w_k'\#\ldots\#w_3'\#w_2'\#w_1'$ **amb** $k \geqslant 2$ **complint que amb un pas de reescriptura sobre cada** w_i **amb** $i < k$ **es pot obtenir** ${w_{i+1}'}^r$. **Tot plegat garantitzarà que tindran intersecció no buida si i només si** v **és accessible des de** u. **Per simplificar suposem que l'alfabet és** $\{a, b\}$. **Llavors** G_1 **és:**

$S \rightarrow u\#X\#u^r$
$X \rightarrow v\#v^r \mid Y$
$Y \rightarrow aYa \mid bYb \mid \#X\#$

Només falta per obtenir la descripció d'una gramàtica G_2 adequada, i això és el que heu de fer (no cal justificació i val 1 punt).

Resposta:

$S \rightarrow S'\#X$
$X \rightarrow aX \mid bX \mid \lambda$
$S' \rightarrow aS'a \mid bS'b \mid u_1S''v_1^r \mid \ldots \mid u_nS''v_n^r$
$S'' \rightarrow aS''a \mid bS''b \mid \#X\# \mid \#S'\#$

(Examen de Gener-2009)

1. (1 punt en total) Contesteu sobre el mateix full de l'enunciat a les qüestions següents.

- (0.25) La concatenació de llenguatges, és simplificable per a llenguatges no buits? És a dir, sabent $L_1L_2 = L_1L_3$ i $L_1 \neq \emptyset$ concloem $L_2 = L_3$? En cas afirmatiu doneu-ne una raó breu; en cas negatiu doneu un contraexemple.

  Resposta: No és simplificable. Tenim un contraexemple definint L_1 com a^*, L_2 com $\{\lambda\}$, i L_3 com $\{\lambda, a\}$.

- (0.25) Doneu un exemple de llenguatge que no sigui semidecidible.

  Reposta: $\overline{\mathtt{K}} = \{x | M_x(x) \uparrow\}$

- (0.5) Sigui $X_1 \to \alpha_1, \ldots, X_n \to \alpha_n$ el conjunt de regles/produccions d'una gramàtica incontextual G que genera mots sobre $\{a, b\}$, i S el seu símbol no-terminal inicial (noteu que els X_i seran no-terminals i els α_i mots sobre terminals i no-terminals). Dibuixeu o descriviu formalment un autòmat amb pila indeterminista que reconegui $\mathtt{L}(G)$.

  Resposta: El PDA té 3 estats q_0, q_1, q_2. El seu estat inicial és q_0, i q_2 n'és l'únic estat acceptador. El seu conjunt de regles conté: $\{q_0 \underset{Z_0|Z_0S}{\to} q_1,\ q_1 \underset{X_1|\alpha_1^R}{\to} q_1,\ \ldots,\ q_1 \underset{X_n|\alpha_n^R}{\to} q_1,\ q_1 \underset{aa|\lambda}{\to} q_1,\ q_1 \underset{bb|\lambda}{\to} q_1,\ q_1 \underset{Z_0|\lambda}{\to} q_2\}$

2. (5.5 punts en total) Recordeu que un morfisme $\sigma : \Sigma^* \to \Sigma^*$ és una funció que es pot definir només per a símbols $\sigma : \Sigma \to \Sigma^*$, i llavors queda definida per a mots per extensió segons $\sigma(a_1 \ldots a_n) = \sigma(a_1) \ldots \sigma(a_n)$.

   Un morfisme σ, com a funció, queda també definit sobre llenguatges amb l'extensió habitual $\sigma(L) = \{\sigma(w) | w \in L\} = \{y | \exists x \in L : (\sigma(x) = y)\}$.

   Sigui $\sigma : \Sigma^* \to \Sigma^*$ un morfisme i $L \subseteq \Sigma^*$ un llenguatge. Definim $\overline{\sigma}(L)$ com el llenguatge dels mots que s'obtenen a base d'agafar un mot u de L, i reemplaçar alguns dels seus símbols (no necessàriament tots i eventualment cap) per la seva imatge per σ. Formalment:

   $$\overline{\sigma}(L) = \{v_1 \cdots v_n | \exists u_1, \ldots, u_n : ((u_1 \cdots u_n) \in L \land \forall i : (v_i = u_i \lor v_i = \sigma(u_i)))\}$$

   Per exemple, si $L = \{ab, b\}$, $\sigma(a) = bb$ i $\sigma(b) = \lambda$, llavors $\sigma(L) = \{bb, \lambda\}$, i en canvi, $\overline{\sigma}(L) = \{ab, bbb, a, bb, b, \lambda\}$.

   Contesteu a les següents preguntes sobre el mateix full de l'enunciat.

   - (0.5) Justifiqueu que, si L és regular, llavors $\sigma(L)$ és regular. Això s'ha fet durant el curs de diverses maneres, i n'hi ha prou que digueu com retocar una expressió regular que genera L, per tal que passi a generar $\sigma(L)$, sense més explicació. (Per simplificar assumiu $\Sigma = \{a, b\}$.)

     Resposta: Sigui E una expressió regular que defineix L. Hem de modificar E reemplaçant simultàniament totes les ocurrències de a per $\sigma(a)$, i totes les ocurrències de b per $\sigma(b)$.

   - (0.75) Justifiqueu que, si L és regular, llavors $\overline{\sigma}(L)$ és regular. Novament, n'hi ha prou que digueu com retocar una expressió regular que genera L, per tal que passi a generar $\overline{\sigma}(L)$, sense més explicació. (Per simplificar assumiu $\Sigma = \{a, b\}$.)

     Resposta: Sigui E una expressió regular que defineix L. Hem de modificar E reemplaçant simultàniament totes les ocurrències de a per $(\sigma(a)+a)$, i totes les ocurrències de b per $(\sigma(b)+b)$.

- (0.5) Justifiqueu que, si L és incontextual, llavors $\sigma(L)$ és incontextual. Això s'ha fet durant el curs de diverses maneres, i n'hi ha prou que digueu com retocar una gramàtica que genera L, per tal que passi a generar $\sigma(L)$, sense més explicació. (Per simplificar assumiu $\Sigma = \{a, b\}$.)

  Resposta: Sigui G una CFG que genera L. Hem de modificar les regles de G reemplaçant simultàniament totes les ocurrències de a per $\sigma(a)$, i totes les ocurrències de b per $\sigma(b)$.

- (0.75) Justifiqueu que, si L és incontextual, llavors $\overline{\sigma}(L)$ és incontextual. N'hi ha prou que digueu com retocar una gramàtica que genera L, per tal que passi a generar $\overline{\sigma}(L)$, sense més explicació. (Per simplificar assumiu $\Sigma = \{a, b\}$.)

  Resposta: Sigui G una CFG que genera L. Ens inventem dos no-terminals nous A i B. Hem de modificar les regles de G reemplaçant simultàniament totes les ocurrències de a per A, i totes les ocurrències de b per B. A més, hem d'afegir les noves regles $A \to a|\sigma(a)$ i $B \to b|\sigma(b)$.

Definim $\sigma^n(L)$ com $\overbrace{\sigma(\cdots(\sigma}^{n)}(L))\ldots)$, i $\overline{\sigma}^n(L)$ com $\overbrace{\overline{\sigma}(\cdots(\overline{\sigma}}^{n)}(L))\ldots)$. Observeu que $\sigma^0(L) = L$, $\sigma^1(L) = \sigma(L)$, $\sigma^2(L) = \sigma(\sigma(L)), \ldots$ i $\overline{\sigma}^0(L) = L$, $\overline{\sigma}^1(L) = \overline{\sigma}(L)$, $\overline{\sigma}^2(L) = \overline{\sigma}(\overline{\sigma}(L)), \ldots$.
Definim $\sigma^*(L)$ com $\bigcup_{n=0}^{\infty}(\sigma^n(L))$ i $\overline{\sigma}^*(L)$ com $\bigcup_{n=0}^{\infty}(\overline{\sigma}^n(L))$

- (0.75) Donat el llenguatge $L = \{ab\}$ i el morfisme definit per $\sigma(a) = ac$, $\sigma(b) = cb$ $\sigma(c) = c$, digueu quins són $\sigma^*(L)$ i $\overline{\sigma}^*(L)$. Escriviu autòmats reconeixent tots dos llenguatges per a justificar que són regulars (només heu de fer el graf dels autòmats, o descriu-re'ls formalment d'alguna manera, però no cal el desenvolupament fins obtenir-los).

  Resposta: $\sigma^*(L)$ és $\{ac^{2n}b | n \geqslant 0\}$, i el seu DFA és:

	a	b	c
q_0	q_1	q_r	q_r
q_1	q_r	q_f	q_2
q_2	q_r	q_r	q_1
q_r	q_r	q_r	q_r
$\dagger q_f$	q_r	q_r	q_r

  $\overline{\sigma}^*(L)$ és $\{ac^n b | n \geqslant 0\}$, i el seu DFA és:

	a	b	c
q_0	q_1	q_r	q_r
q_1	q_r	q_f	q_1
q_r	q_r	q_r	q_r
$\dagger q_f$	q_r	q_r	q_r

- (0.75) Quan un llenguatge L és regular, resulta que $\sigma^*(L)$ no té per què ser necessàriament regular, i ni tan sols incontextual. Doneu un exemple on L sigui regular, i en canvi $\sigma^*(L)$ no sigui regular (0.5 punts en tal cas) i, si pot ser, que ni tan sols sigui incontextual (0.75 punts en tal cas). Digueu quin és el $\sigma^*(L)$ resultant, i si és o no incontextual, però no cal que justifiqueu la no-regularitat o no-incontextualitat de $\sigma^*(L)$.

  Resposta: $L = \{a\}$ és regular, però definint $\sigma(a) = aa$, resulta que $\sigma^*(L) = \{a^{2^n} | n \geqslant 0\}$ no és incontextual.

Resposta alternativa: $L = \{a'b'c'\}$ és regular, però definint $\sigma(a') = aa'$, $\sigma(b') = bb'$, $\sigma(c') = cc'$, $\sigma(a) = a$, $\sigma(b) = b$ i $\sigma(c) = c$, resulta que $\sigma^*(L) = \{a^n a' b^n b' c^n c' | n \geqslant 0\}$ no és incontextual.

- (0.5) Quan un llenguatge L és regular, resulta que $\overline{\sigma}^*(L)$ no té per què ser necessàriament regular. Doneu un exemple on L sigui regular, i en canvi $\overline{\sigma}^*(L)$ no ho sigui. Digueu quin és el $\overline{\sigma}^*(L)$ resultant, però no cal que justifiqueu la no-regularitat de $\overline{\sigma}^*(L)$.

  Resposta: $L = \{c\}$ és regular, però definint $\sigma(a) = a$, $\sigma(b) = b$ i $\sigma(c) = acb$, resulta que $\overline{\sigma}^*(L) = \{a^n c b^n | n \geqslant 0\}$ no és regular.

- (1) Justifiqueu que, si L és incontextual, llavors $\overline{\sigma}^*(L)$ també ho és. Només cal que indiqueu com modificar una gramàtica G que genera L de manera que passi a generar $\overline{\sigma}^*(L)$. No cal que demostreu que la construcció funciona, però expliqueu almenys quina és la idea intuïtiva que hi ha al darrera. (Per simplificar assumiu $\Sigma = \{a, b\}$.)

  Resposta: Sigui G una CFG que genera L. Ens inventem dos no-terminals nous A i B. Sigui w_a la paraula que s'obté a partir de $\sigma(a)$ reemplaçant simultàniament totes les ocurrències de a per A, i totes les ocurrències de b per B. Anàlogament, sigui w_b la paraula que s'obté a partir de $\sigma(b)$ reemplaçant simultàniament totes les ocurrències de a per A, i totes les ocurrències de b per B.
  Hem de modificar les regles de G reemplaçant simultàniament totes les ocurrències de a per A, i totes les ocurrències de b per B. Ademés, hem d'afegir les noves regles $A \to a | w_a$ i $B \to b | w_b$.
  La idea és que les noves regles permeten aplicar el morfisme sobre cada símbol inicial o que aparegui posteriorment. Si sobre un símbol concret no el volem aplicar, llavors reescrivim directament a a o b, depenent del cas.

3. (2 punts) Classifiqueu com a decidible, semidecidible però no decidible, o no semidecidible, el problema següent: $C = \{\langle x, y\rangle \mid \exists z, w : (M_x(z) \downarrow \wedge M_y(w) \downarrow \wedge M_x(z) = M_y(w))\}$

Resposta Demostrem que és indecidible reduint des de `K`. La reducció consisteix a, per cada x d'entrada, generar el parell $\langle p(x), q\rangle$, on $p(x)$ és un número que codifica el programa

```
entrada y
   Simular Mx(x)
   sortida 1
```

i q és un número que codifica el programa:

```
entrada y
   sortida 1
```

Si x pertany a `K`, llavors $\mathtt{M}_x(x) \downarrow$, i per tant el programa codificat per $p(x)$ dona sempre sortida 1. Com que el programa codificat per q dona sempre sortida 1, resulta que el parell $\langle p(x), q\rangle$ és del llenguatge C.

Si x no pertany a `K`, llavors $\mathtt{M}_x(x) \uparrow$, i per tant el programa codificat per $p(x)$ no s'atura per a cap entrada. Així doncs, el parell $\langle p(x), q\rangle$ no és del llenguatge C.

Ara justificarem que el llenguatge anterior és semi-decidible. N'hi ha prou amb mostrar un programa que semi-decideix el conjunt.

```
entrada ⟨x, y⟩
   t:=1
   Executar indefinidament:
      Per a cada i, j entre 0 i t fer:
         Si M_x(i) i M_y(j) s'aturen en t passos i M_x(i) = M_y(j)
            llavors acceptar
      t:=t + 1
```

4. (1.5 punts) Considereu els següents tres problemes:

   - (intersecció no buida) Donades dues gramàtiques, saber si tenen intersecció no buida.

   $$\{\langle G_1, G_2\rangle \mid \mathtt{L}(G_1) \cap \mathtt{L}(G_2) \neq \emptyset\}$$

   - (3-and-intersecció no buida)

   $$\{\langle G_1, G_2, G_3\rangle \mid \mathtt{L}(G_1) \cap \mathtt{L}(G_2) \neq \emptyset \wedge \mathtt{L}(G_1) \cap \mathtt{L}(G_3) \neq \emptyset \wedge \mathtt{L}(G_2) \cap \mathtt{L}(G_3) \neq \emptyset\}$$

   - (3-or-intersecció no buida)

   $$\{\langle G_1, G_2, G_3\rangle \mid \mathtt{L}(G_1) \cap \mathtt{L}(G_2) \neq \emptyset \vee \mathtt{L}(G_1) \cap \mathtt{L}(G_3) \neq \emptyset \vee \mathtt{L}(G_2) \cap \mathtt{L}(G_3) \neq \emptyset\}$$

   (a) (0.75) Reduïu el primer problema al segon. Justifiqueu la resposta.

   (b) (0.75) Reduïu el primer problema al tercer. Justifiqueu la resposta.

   Resposta:
   Per simplificar, suposem que l'alfabet és $\Sigma = \{a, b\}$.

   - **Una reducció possible és: $\langle G_1, G_2\rangle \mapsto \langle G_1, G_2, S \to Sa|Sb|\lambda\rangle$. Està clar que si $\langle G_1, G_2, S \to Sa|Sb|\lambda\rangle$ satisfà 3-and-intersecció no buida, llavors G_1 i G_2 generen algun mot comú. Per l'altra direcció, està clar que si G_1 i G_2 generen algun mot comú, aquest també serà generat amb $S \to Sa|Sb|\lambda$, ja que genera Σ^*.**
   - **Una reducció possible és: $\langle G_1, G_2\rangle \mapsto \langle G_1, G_2, S \to S\rangle$. Està clar que si G_1 i G_2 generen algun mot comú, llavors $\langle G_1, G_2, S \to S\rangle$ satisfà 3-or-intersecció no buida. Per l'altra direcció, està clar que ni G_1 ni G_2 generen mots comuns amb $S \to S$, ja que aquesta CFG genera el llenguatge buit. Per tant, si $\langle G_1, G_2, S \to S\rangle$ satisfà 3-or-intersecció no buida, llavors G_1 i G_2 generen algun mot comú.**

(Examen de Juny-2009)

1. (3 punts) Raoneu la veracitat o falsedat de les afirmacions següents. En cas afirmatiu cal escriure una demostració, on podeu donar per sabudes totes les propietats de tancament de llenguatges que s'han vist a l'assignatura. En cas negatiu cal donar un contraexemple, i una explicació del contraexemple si aquest no és prou evident. (Cada apartat val 0.5 punts.)

(a) Si $L \subseteq \Sigma^*$ és no regular i $\sigma : \Sigma^* \to \Sigma^*$ és un morfisme, llavors $\sigma(L)$ és no regular.

(b) Si $L \subseteq \Sigma^*$ és no regular i $\sigma : \Sigma^* \to \Sigma^*$ és un morfisme, llavors $\sigma^{-1}(L)$ és no regular.

(c) Si L és no regular, llavors $\overline{L}$ és no regular.

(d) Si L és no regular, llavors L^R és no regular.

(e) Si L és no incontextual, llavors $\overline{L}$ és no incontextual.

(f) Si L és no incontextual, llavors L^R és no incontextual.

Resposta:

(a) No es cumpleix l'afirmació. Contraexemple: Definint $\Sigma = \{a, b\}, \sigma(a) = \lambda, \sigma(b) = \lambda$, resulta que $L = \{a^n b^n | n \geqslant 0\}$ és no regular però $\sigma(L) = \{\lambda\}$ sí és regular.

(b) No es cumpleix l'afirmació. Contraexemple: Definint $\Sigma = \{a, b\}, \sigma(a) = \lambda, \sigma(b) = \lambda$, resulta que $L = \{a^n b^n | n \geqslant 0\}$ és no regular però $\sigma^{-1}(L) = \{a, b\}^*$ sí és regular.

(c) Sí es cumpleix l'afirmació. Per contrarecíproc, l'enunciat "Si L és no regular, llavors $\overline{L}$ és no regular" és equivalent a l'enunciat "Si $\overline{L}$ és regular, llavors L és regular", que sabem que és cert perquè els llenguatges regulars són tancats per complementari i $\overline{\overline{L}}$ és L.

(d) Sí es cumpleix l'afirmació. Per contrarecíproc, l'enunciat "Si L és no regular, llavors L^R és no regular" és equivalent a l'enunciat "Si L^R és regular, llavors L és regular", que sabem que és cert perquè els llenguatges regulars són tancats per revessat i $(L^R)^R$ és L.

(e) No es cumpleix l'afirmació. És fàcil obtenir una gramàtica pel llenguatge $L = \{a^i b^j c^k | i \neq j \vee j \neq k\}$. Però el seu complementari $\overline{L}$ no és incontextual, cosa que justifiquem a continuació. Suposem que $\overline{L}$ és incontextual. Llavors, la seva intersecció amb el llenguatge regular $a^* b^* c^*$ és incontextual. Però aquesta intersecció és $\{a^n b^n c^n | n \geqslant 0\}$, que no és incontextual: contradicció.

Resposta alternativa:

No es cumpleix l'afirmació. Per contrarecíproc, l'enunciat "Si L és no incontextual, llavors $\overline{L}$ és no incontextual." és equivalent a l'enunciat "Si $\overline{L}$ és incontextual, llavors L és incontextual". Per tant, si l'enunciat fos cert, llavors els llenguatges incontextuals serien tancats per complementari. Donat que són tancats per unió, llavors també ho serien per intersecció, ja que $A \cap B = \overline{\overline{A} \cup \overline{B}}$. Per tant, n'hi ha prou amb que donem un contraexemple al tancament dels llenguatges incontextuals per intersecció, i aquest és $\{a^n b^n c^m | n, m \geqslant 0\} \cap \{a^m b^n c^n | n, m \geqslant 0\} = \{a^n b^n c^n | n \geqslant 0\}$, ja que $\{a^n b^n c^m | n, m \geqslant 0\}$ i $\{a^m b^n c^n | n, m \geqslant 0\}$ són incontextuals, però $\{a^n b^n c^n | n \geqslant 0\}$ no ho és.

(f) Sí es cumpleix l'afirmació. Per contrarecíproc, l'enunciat "Si L és no incontextual, llavors L^R és no incontextual" és equivalent a l'enunciat "Si L^R és incontextual, llavors L és incontextual", que sabem que és cert perquè els llenguatges incontextuals són tancats per revessat i $(L^R)^R$ és L.

2. (3 punts) Resoleu els exercicis següents. (Cada apartat val 0.75 punts.)

(a) Obtingueu un DFA mínim per al llenguatge sobre $\{a, b\}$ dels mots amb un nombre parell de a's i un nombre senar de b's, és a dir: $\{w \in \{a, b\}^* | \ |w|_a \in \dot{2} \wedge |w|_b \in (\dot{2} + 1)\}$. Justifiqueu la seva minimalitat.

En cas que no obtingueu el DFA per descomposició en llenguatges més simples, cal que el justifiqueu, explicant almenys el significat de cada estat.

(b) Sigui A el DFA que heu construït a l'apartat anterior. Siguin $q_1, \ldots, q_N$ els estats de A, i sigui F el conjunt d'estats acceptadors de A. Per cada estat q_i de A, sigui L_{q_i} el llenguatge reconegut començant l'execució des de q_i, és a dir: $L_{q_i} = \{w | q_i w \in F\}$. Escriviu un conjunt de N equacions de llenguatges, amb N variables $X_1, \ldots, X_N$, que tingui com a única solució $X_1 := L_{q_1}, \ldots, X_N := L_{q_N}$.

En aquest apartat no calen justificacions.

(c) Raoneu per què el conjunt d'equacions de l'apartat anterior té solució única.

Si voleu, podeu escriure un recordatori de què diu el Lema d'Arden sobre equacions del tipus $X = AX + B$ i utilitzar-ho.

(d) Obtingueu una expressió regular per a $\{w \in \{a,b\}^* | \ |w|_a \in \dot{2} \wedge |w|_b \in (\dot{2}+1)\}$.

Si no l'obteniu resolent el conjunt d'equacions anterior, llavors cal que doneu una justificació de per què l'expressió regular proposada representa el llenguatge.

Resposta:

(a) **Construïm un DFA amb 4 estats $q_{00}, q_{01}, q_{10}, q_{11}$. El significat dels estats és el següent. Si havent llegit un mot w arribem a l'estat q_{ij}, llavors el nombre d'a's de w és i mòdul 2, i el nombre de b's de w és j mòdul 2.**

	a	b
$\rightarrow q_{00}$	q_{10}	q_{01}
$\dagger q_{01}$	q_{11}	q_{00}
q_{10}	q_{00}	q_{11}
q_{11}	q_{01}	q_{10}

Per veure que el DFA és mínim, podem aplicar l'algorisme de minimització i veure que no canvia, o bé argumentar que tots els estats són distingibles. Procedim d'aquesta segona manera.

El mot b ens distingeix q_{00} de q_{01}, perquè $q_{00}b$ és q_{01}, que és acceptador, i $q_{01}b$ és q_{00}, que és rebutjador. Similarment, b ens distingeix q_{00} de q_{10}, i q_{00} de q_{11}. El mot λ distingeix q_{01} de q_{10}, i q_{01} de q_{11}. El mot a ens distingeix q_{10} de q_{11}.

(b)

$$
\begin{aligned}
X_{00} &= aX_{10} + bX_{01} \\
X_{01} &= aX_{11} + bX_{00} + \Lambda \\
X_{10} &= aX_{00} + bX_{11} \\
X_{11} &= aX_{01} + bX_{10}
\end{aligned}
$$

(c) **El Lema d'Arden ens parla d'equacions $X = AX + B$, on A, B són llenguatges fixats, i X és una variable amb rang en els llenguatges. El lema ens diu que A^*B és una solució d'aquesta equació, i que si A no conté λ, llavors A^*B n'és l'única solució.**

Una equació de la forma $X = AX + B_1Y_1 + \cdots + B_nY_n$ no és com les del Lema d'Arden, perquè hi ha més variables, però sí el podem aplicar quan pensem en valors fixats substituïnt aquestes variables. Així doncs, si A no conté λ, per cada valor de $Y_1, \ldots, Y_n$ el valor de X queda univocament determinat per $A^*(B_1Y_1 + \cdots + B_nY_n)$. És a dir que qualsevol solució de $X = AX + B_1Y_1 + \cdots + B_nY_n$ també cumpleix $X = A^*(B_1Y_1 + \cdots + B_nY_n)$.

En l'apartat següent veurem que, sempre que apliquem el Lema d'Arden, els A's involucrats no contenen λ. Per tant, la solució a la que arribarem serà, de fet, única.

(d) **Substituïnt la segona i tercera equacions en la primera i quarta, simultàniament, obtenim:**

$$
\begin{aligned}
X_{00} &= a(aX_{00} + bX_{11}) + b(aX_{11} + bX_{00} + \Lambda) \\
X_{11} &= a(aX_{11} + bX_{00} + \Lambda) + b(aX_{00} + bX_{11}) = \\
&= (aa + bb)X_{11} + (ab + ba)X_{00} + a
\end{aligned}
$$

Aplicant el Lema d'Arden a l'última d'aquestes equacions obtenim:

$X_{11} = (aa+bb)^*((ab+ba)X_{00}+a) = (aa+bb)^*(ab+ba)X_{00}+(aa+bb)^*a$

I susbtituïnt en l'altra equació obtenim:

$$X_{00} = a(aX_{00}+b(aa+bb)^*(ab+ba)X_{00}+b(aa+bb)^*a)+ \\ +b(a(aa+bb)^*(ab+ba)X_{00}+a(aa+bb)^*a+bX_{00}+\Lambda)$$

Agrupant termes ens queda:

$$X_{00} = \big(aa+ab(aa+bb)^*(ab+ba)+ba(aa+bb)^*(ab+ba)+bb\big)X_{00}+ \\ +\big(ab(aa+bb)^*a+ba(aa+bb)^*a+b\big)$$

Aplicant novament el Lema d'Arden concloem:

$$X_{00} = \big(aa+ab(aa+bb)^*(ab+ba)+ba(aa+bb)^*(ab+ba)+bb\big)^* \\ \big(ab(aa+bb)^*a+ba(aa+bb)^*a+b\big)$$

Donat que volem expressar els mots acceptats començant l'execució des de l'estat inicial q_{00}, aquesta última expressió és la que buscàvem.

3. (1.5 punts) Classifiqueu com a decidible, semi-decidible però no decidible, o no semi-decidible, el problema següent: $C = \{\langle x,y\rangle \mid \forall z,w : (M_x(z)\downarrow \wedge M_y(w)\downarrow \wedge M_x(z) > M_y(w))\}$

Resposta:

Demostrem que C no és ni semi-decidible reduïnt des de $\overline{K}$. Per cada nombre natural x, la reducció dona com a sortida la parella $\langle p_1,p_2\rangle$, on p_1 és el natural que codifica el programa següent:

Entrada y
  Si $M_x(x)$ s'atura en y o menys passos llavors sortida 1
  Sino sortida 3

I p_2 és el natural que codifica el programa següent:

Entrada y
  Sortida 2

Donat un x de $\overline{K}$, sabem que $M_x(x)\uparrow$. Per tant, el programa p_1 dona sortida 3 per tota entrada, i el programa p_2 dona sortida 2 per tota entrada, de manera que la parella $\langle p_1,p_2\rangle$ és de C.

Donat un x que no és de $\overline{K}$, sabem que $M_x(x)\downarrow$. Sigui T el temps que triga $M_x(x)$ en aturar. Llavors, el programa p_1 dona sortida 1 per a entrada T, i el programa p_2 dona sortida 2 per a entrada T, de manera que la parella $\langle p_1,p_2\rangle$ no és de C.

4. (2.5 punts en total) Considereu els problemes següents.

- (accessibilitat de mots) Donada una llista finita R de regles de reescriptura (un conjunt finit de parells (u_i,v_i) de mots diferents de λ) i dos mots u (inicial), v (final), determinar si és possible passar de u a v utilitzant les regles de reescriptura (un seguit de substitucions d'algun submot u_i pel seu corresponent v_i).
$$\{\langle R=\{u_1\to v_1,\ldots,u_n\to v_n\},u,v\rangle \mid \quad u\to_R^* v \wedge \\ \forall i\in\{1,\ldots,n\} : (|u_i|,|v_i|\neq 0)\}$$
- (accessibilitat de símbol especial) Donada una llista finita R de regles de reescriptura (un conjunt finit de parells (u_i,v_i) de mots diferents de λ) i un mot u (inicial), i un símbol $\#$ (final) que no apareix en cap u_i, determinar si és possible passar de u a $\#$ utilitzant les regles de reescriptura.
$$\{\langle R=\{u_1\to v_1,\ldots,u_n\to v_n\},u,\#\rangle \mid \quad u\to_R^* \# \wedge \forall i\in\{1,\ldots,n\} : \\ (|u_i|,|v_i|\neq 0 \wedge |u_i|_\# = 0)\}$$

- (acabament) Donada una llista finita R de regles de reescriptura (un conjunt finit de parells (u_i, v_i) de mots diferents de λ) i un mot u (inicial), determinar si existeix un mot v tal que $u \rightarrow_R^* v$ i v no pot ser reescrit per cap regla (és a dir, v és accessible des de u, però no admet cap derivació ulterior, i per tant existeix una derivació des de u que acaba).

$$\{\langle R = \{u_1 \to v_1, \ldots, u_n \to v_n\}, u\rangle | \quad \exists v : (u \rightarrow_R^* v \wedge \nexists w : (v \rightarrow_R w)) \\ \wedge \forall i \in \{1, \ldots, n\} : (|u_i|, |v_i| \neq 0)\}$$

(a) (0.75 punts) Reduïu el primer problema (accessibilitat de mots) al segon (accessibilitat de símbol especial). Justifiqueu correctament la vostra reducció. (una reducció correcta val 0.5, i la justificació els altres 0.25).

(b) (0.75 punts) Demostreu que la reducció següent del segon problema (accessibilitat de símbol especial) al tercer (acabament) no és correcta, donant un contraexemple: $\langle R, u, \# \rangle \mapsto \langle R, u \rangle$.

(c) (1 punt) Reduïu el segon problema (accessibilitat de símbol especial) al tercer (acabament). Justifiqueu correctament la vostra reducció. (Una reducció correcta val 0.5, i la justificació els altres 0.5).

Resposta:

(a) Proposem la següent reducció:

$\langle R, u, v \rangle \mapsto \langle R \cup \{\$v\$ \to \#\}, \$u\$, \# \rangle$ on $\$, \#$ són símbols nous.

Clarament, la transformació és computable i total.

Sigui $\langle R, u, v \rangle$ una entrada d'accessibilitat de mots, i $\langle R \cup \{\$v\$ \to \#\}, \$u\$, \# \rangle$ el resultat de la seva transformació.

Suposem primer que $\langle R, u, v \rangle$ té resposta afirmativa en el problema d'accessibilitat de mots, és a dir, tots els u_i, v_i són diferents de λ i $u \rightarrow_R^* v$. Llavors, també passa que tots els u_i, v_i, juntament amb $\$v\$, \#$ són no buits (encara que v pugui ser buit), i $\#$ no apareix a la part esquerra de cap de les regles de $R \cup \{\$v\$ \to \#\}$. Ademés, tenim $\$u\$ \rightarrow_R^* \$v\$ \rightarrow_{\$v\$ \to \#} \#$. Per tant, $\langle R \cup \{\$v\$ \to \#\}, \$u\$, \# \rangle$ té resposta afirmativa en el problema d'accessibilitat de símbol especial.

Ara, suposem que $\langle R \cup \{\$v\$ \to \#\}, \$u\$, \# \rangle$ té resposta afirmativa en el problema d'accessibilitat de símbol especial. Llavors, tots els u_i, v_i són diferents de λ. Ademés, existeix una derivació $\$u\$ \rightarrow_{R \cup \$v\$ \to \#}^* \#$. Donat que $\#$ només apareix a la regla $\$v\$ \to \#$ i que no apareix en $\$u\$$, aquesta regla s'ha fet servir una i només una vegada en la derivació $\$u\$ \rightarrow_{R \cup \$v\$ \to \#}^* \#$. Per tant, aquesta derivació és de la forma $\$u\$ \rightarrow_R^* w_1\$v\$w_2 \rightarrow_{\$v\$ \to \#} w_1\#w_2 \rightarrow_R^* \#$, i conté subderivations $\$u\$ \rightarrow_R^* w_1\$v\$w_2$, $w_1 \rightarrow_R^* \lambda$ i $w_2 \rightarrow_R^* \lambda$. Però com que tots els costats de regla són no buits, λ no és generable amb un o més passos, de manera que w_1 i w_2 són λ. Per tant, tenim una derivació $\$u\$ \rightarrow_R^* \$v\$$, i com que $\$$ no apareix a R, tenim una derivació $u \rightarrow_R^* v$. Això conclou que $\langle R, u, v \rangle$ té resposta afirmativa en el problema d'accessibilitat de mots.

Resposta alternativa d'aquest apartat:

La transformació a partir de $\langle R = \{u_1 \to v_1, \ldots, u_n \to v_n\}, u, v \rangle$ procedeix així:

- **Si tots els $|u_i|, |v_i|$ són diferents de 0 i $u = v = \lambda$, llavors generem $\langle R = \emptyset, \#, \# \rangle$.**
- **En cas contrari, generem $\langle R \cup \{v \to \#\}, u, v \rangle$, on $\#$ és un símbol nou.**

Clarament, la transformació és computable i total.

Sigui $\langle R = \{u_1 \to v_1, \ldots, u_n \to v_n\}, u, v \rangle$ una instància d'accessibilitat de mots amb resposta afirmativa. Llavors, tots els $|u_i|, |v_i|$ són diferents de 0. En el cas en que v i

u són λ, generem $\langle R = \emptyset, \#, \#\rangle$, que té resposta afirmativa en accessibilitat de símbol especial. Suposem que o bé u o bé v no és λ. Llavors cap dels dos és λ, perquè $u \to_R^* v$ i cap de les regles de R té λ com a part esquerra o dreta. En aquest cas estem generant $\langle R \cup \{v \to \#\}, u, v\rangle$. Donat que $u \to_R^* v$, deduïm $u \to_R^* v \to_{v\to\#} \#$. Además, com que $\#$ és nou, no apareix a l'esquerra de cap regla de $u \to_R^* v \to_{v\to\#} \#$. Recalquem además que tots els $|u_i|, |v_i|$ no són 0, que $|v|$ no és 0 (per ser diferent de λ), i que $|\#|$ no és 0. Per tant,$\langle R \cup \{v \to \#\}, u, v\rangle$ té resposta afirmativa en accessibilitat de símbol especial.

Sigui $\langle R = \{u_1 \to v_1, \ldots, u_n \to v_n\}, u, v\rangle$ una instància d'accessibilitat de mots amb resposta negativa. En tal cas no es cumpleix simultàniament que tots els $|u_i|, |v_i|$ siguin diferents de 0 i $u = v = \lambda$. Per tant, hem generat $\langle R \cup \{v \to \#\}, u, v\rangle$. En cas que algun $|u_i|$ o $|v_i|$ sigui 0, també $\langle R \cup \{v \to \#\}, u, v\rangle$ té resposta negativa en accessibilitat de símbol especial. Suposem que tots els $|u_i|, |v_i|$ són diferents de 0. Llavors o bé u o bé v no són λ. Si v és λ, llavors no es cumpleix $|v| \neq 0$, de manera que $\langle R \cup \{v \to \#\}, u, v\rangle$ té resposta negativa en accessibilitat de símbol especial. Si u és λ, llavors v no és λ, i no és possible accedir de u a $\#$ amb regles que no tenen λ com a part dreta d'alguna regla, de manera que $\langle R \cup \{v \to \#\}, u, v\rangle$ té resposta negativa en accessibilitat de símbol especial. Suposem, finalment, que ni u ni v és λ. Ens queda per veure que no és possible accedir de u a $\#$ utilitzant $R' = R \cup \{v \to \#\}$. Ho veiem per reducció a l'absurd. Si existís una derivació $u \to_{R'}^* \#$, llavors, com que $\#$ no apareix ni en u, ni a l'esquerra de cap regla, resulta que la regla $v \to \#$ s'ha utilitzat una i només una vegada. Per tant, la derivació $u \to_{R'}^* \#$ és de la forma $u \to_R^* w_1 v w_2 \to_{v\to\#} w_1\#w_2 \to_R^* \#$. Així doncs, tenim subderivacions de la forma $u \to_R^* w_1 v w_2$, $w_1 \to_R^* \lambda$ i $w_2 \to_R^* \lambda$. Per les condicions sobre R, les derivacions $w_1 \to_R^* \lambda$ i $w_2 \to_R^* \lambda$ només poden ser buides, de manera que $w_1 = w_2 = \lambda$. Això conclou $u \to_R^* v$, lo qual està en contradicció amb el fet que $\langle R, u, v\rangle$ té resposta negativa en el problema d'accessibilitat de mots.

(b) Aquesta reducció ens transforma $\langle\{a \to b\}, a, \#\rangle$ en $\langle\{a \to b\}, a\rangle$. El primer té resposta negativa en accessibilitat de símbol especial, però afirmativa en acabament.

(c) La transformació $\langle R, u, \#\rangle \mapsto \langle R \cup \{\$ \to \$,\ \$\#\$ \to \&\},\ \$u\$\rangle$, on $\$$ i $\&$ són símbols nous, ens redueix accessibilitat de símbol especial a acabament.

Clarament, la transformació és computable i total.

Sigui $\langle R = \{u_1 \to v_1, \ldots, u_n \to v_n\}, u, \#\rangle$ una entrada d'accessibilitat de símbol especial i $\langle R \cup \{\$ \to \$,\ \$\#\$ \to \&\},\ \$u\$\rangle$ el resultat de la seva transformació.

Suposem que $\langle R = \{u_1 \to v_1, \ldots, u_n \to v_n\}, u, \#\rangle$ té resposta afirmativa en el problema d'accessibilitat de símbol especial. Llavors, tots els $|u_i|, |v_i|$ no són 0 i es cumpleix $u \to_R^* \#$. Per tant, $\$u\$ \to_R^* \$\#\$ \to_{\$\#\$\to\&} \&$, i donat que $\&$ no pot ser reescrit per cap regla, i que $|\$u\$|, |\$|, |\&|$ no són 0, resulta que $\langle R \cup \{\$ \to \$,\ \$\#\$ \to \&\},\ \$u\$\rangle$ té resposta afirmativa en el problema d'acabament.

Ara, suposem que $\langle R \cup \{\$ \to \$,\ \$\#\$ \to \&\},\ \$u\$\rangle$ té resposta afirmativa en el problema d'acabament. Llavors, tots els $|u_i|, |v_i|$ no són 0, i hi ha alguna derivació que acaba, començant des de $\$u\$$ amb les regles de $R' = R \cup \{\$ \to \$,\ \$\#\$ \to \&\}$. Degut a l'existencia de la regla $\$ \to \$$ i el fet que aquesta derivació acaba, en algun moment s'ha d'haver aplicat $\$\#\$ \to \&$. Fins que s'aplica, els mots intermitjos sempre tenen la forms $\$w\$$, on w és un mot sobre l'alfabet original. I un cop s'aplica ens queda $\&$, que no es pot reescriure més. Además, podem suposar que, en aquesta derivació no s'hi ha aplicat cap vegada la regla $\$ \to \$$, doncs podem eliminar les aplicacions d'aquesta regla sense modificar el fet d'arribar a $\&$. Per tant, la derivació és de la forma $\$u\$ \to_R^* \$\#\$ \to_{\$\#\$\to\&} \&$. La subderivació $\$u\$ \to_R^* \$\#\$$ permet concloure $u \to_R^* \#$, de manera que $\langle R, u, \#\rangle$ té resposta afirmativa en accessibilitat de símbol especial.

Resposta alternativa simplificada:
La reducció $\langle R, u, \#\rangle \mapsto \langle R \cup \{a \to a | a \in \Sigma - \{\#\}\} \cup \{\#\# \to \#\#\}, u\rangle$ també funciona.

(Examen de Gener-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:

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

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 $L_a = \bigcup_{n\geqslant 0}(a+b)^n a(a+b)^* c(a+b)^n$ i $L_b = \bigcup_{n\geqslant 0}(a+b)^n b(a+b)^* c(a+b)^n$.

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	$q_0 q_1$
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
$\to \dagger q_0$	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) $\{x \mid M_x(x) \uparrow\}$ $\{x \mid \forall y : M_x(y) \downarrow\}$
   b) $\{x \mid M_x(x) \downarrow\}$ $\{\langle u, v, R\rangle \mid u \to_R^* v\}$
   c) $\{a^n b^n c^n \mid n \geqslant 0\}$ $\{ww \mid w \in \{a,b\}^*\}$
   d) $\{a^n b^n \mid n \geqslant 0\}$ $\{ww^R \mid w \in \{a,b\}^*\}$
   e) $\{a,b\}^*$ $\{w \in \{0,1\}^* \mid \mathtt{valor}_2(w) \in \dot{3}\}$

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 $1 \leqslant i_1, \ldots, i_k \leqslant n$ amb $k \geqslant 1$ 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 $1 \leqslant i_1, \ldots, i_k \leqslant n$ amb $k \geqslant 1$ 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 unilíteral 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.

(Examen de Juny-2010)

Tingueu present en els dos primers exercicis que les respostes s'han de justificar. La mera presentació d'una proposta no és suficient.

1. (3 punts) Considereu el llenguatge L_1 format pels mots sobre l'alfabet $\{a, b\}$ tals que concatenats amb el seu revessat inclouen el submot $ababb$. Formalment,

$$L_1 = \{w \in (a+b)^* \mid \exists x, y \;\; ww^{\mathsf{R}} = xababby\}.$$

a) Doneu una expressió regular per a aquest llenguatge.

b) Construïu el mínim DFA que el reconeix.

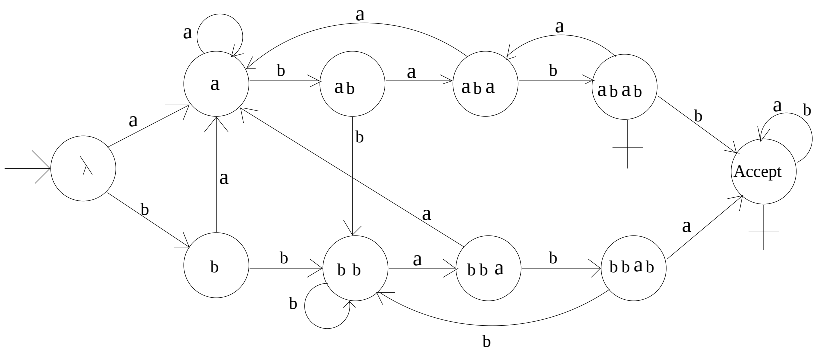
Resposta:

a) Donat un mot w, $ababb$ apareix a ww^{R} si o bé (i) $ababb$ apareix a w, o bé (ii) $ababb$ apareix a w^{R}, o bé (iii) $ababb$ es pot partir en dos mots $xy = ababb$ no buits tals que x és sufix de w i y és prefix de w^{R}.

El cas (ii) és equivalent a imposar que $bbaba$ apareix a w. El cas (iii) és equivalent a imposar que $abab$ és sufix de w. Per tant, la següent expressió regular defineix L_1:

$$(a+b)^* ababb (a+b)^* \; + \; (a+b)^* bbaba (a+b)^* \; + \; (a+b)^* abab$$

b) **D'acord amb l'apartat anterior, el nostre DFA ha de comprovar que el mot d'entrada o bé conté** *ababb*, **o bé conté** *babaa*, **o bé acaba en** *abab*. **Per fer tals comprovacions, cal que recordi el màxim sufix del mot llegit fins el moment tal que és prefix de o bé** *ababb*, **o bé** *babaa*, **o bé** *abab*. **A més a més, ha de recordar si o bé** *ababb* **o bé** *babaa* **ja han aparegut amb anterioritat, doncs en tal cas hem de romandre en estat acceptador. Amb aquestes idees obtenim el següent DFA:**



L'estat `Accept` **cumpleix que un mot d'entrada** w **ens porta fins** `Accept` **si i només si** w **conté** *ababb* **o** *bbaba*. **Els estats que tenen per nom un mot** x **són aquells que cumpleixen el següent: un mot** w **ens porta fins a l'estat** x **si i només si** w **no conté ni** *ababb* **ni** *bbaba*, **i a més a més** x **és el màxim sufix de** w **tal que és prefix de o bé** *ababb* **o bé** *bbaba*. **Aquestes propietats es poden justificar per inducció sobre el nombre de passos d'execució, i justifiquen la correctesa de l'autòmat.**

Per veure que és mínim, apliquem l'algorisme de minimització:

0-**indistingibles:** $\{\lambda, a, ab, aba, b, bb, bba, bbab\}, \{abab, \texttt{Accept}\}$

1-**indistingibles:** $\{\lambda, a, ab, b, bb, bba\}, \{aba\}, \{bbab\}, \{abab\}\{\texttt{Accept}\}$

2-**indistingibles:** $\{\lambda, a, b, bb\}, \{ab\}, \{bba\}, \{aba\}, \{bbab\}, \{abab\}\{\texttt{Accept}\}$

3-**indistingibles:** $\{\lambda, b\}, \{a\}, \{bb\}, \{ab\}, \{bba\}, \{aba\}, \{bbab\}, \{abab\}\{\texttt{Accept}\}$

4-**indistingibles:** $\{\lambda\}, \{b\}, \{a\}, \{bb\}, \{ab\}, \{bba\}, \{aba\}, \{bbab\}, \{abab\}\{\texttt{Accept}\}$

2. (2,5 punts) Doneu una gramàtica que generi els palíndroms de longitud parella que contenen el submot *ababb*. Formalment, es tracta del llenguatge

$$L_2 = \{ww^{R} \mid \exists x, y \in (a+b)^* \;\; ww^{R} = xababby\}.$$

Resposta: Aprofitant l'exercici anterior, L_2 **també es pot definir així:**

$$L_2 = \{ww^{R} \mid w \in L_1\}$$

Per tant, en base al problema anterior, la CFG ha de generar mots palíndroms de longitud parella tals que la primera meitat o bé conté *ababb*, **o bé conté** *bbaba*, **o bé acaba en** *abab*. **D'aquesta manera obtenim la següent CFG:**

$$\begin{array}{lcl} S & \rightarrow & aSa \mid bSb \mid ababbXbbaba \mid bbabaXababb \mid ababbaba \\ X & \rightarrow & aXa \mid bXb \mid \lambda \end{array}$$

S **genera** L_2, **i** X **genera els mots palíndroms de longitud parella.**

3. (1,5 punts) Classifiqueu com a decidibles o no decidibles els problemes següents. Raoneu-ne les respostes.

1. Determinar si un DFA donat reconeix el conjunt de mots que inclouen el submot *ababb*.
2. Determinar si un DPDA donat reconeix el conjunt de mots que inclouen el submot *ababb*.
3. Determinar si una TM donada reconeix el conjunt de mots que inclouen el submot *ababb*.

Resposta:

1. **Sigui A el DFA mínim que reconeix el conjunt de mots que inclouen el submot *ababb* (A existeix perque aquest llenguatge és regular). Per a decidir si un DFA B d'entrada reconeix aquest llenguatge, n'hi ha prou amb minimitzar B i comprovar si el resultat és isomorf a A. En cas afirmatiu B també reconeix aquest llenguatge. En cas negatiu, B no reconeix aquest llenguatge. Aquest algorisme funciona perque sabem que l'autòmat mínim reconeixent un llenguatge regular és únic, tret de renombrament d'estats (isomorfisme).**
2. **Sigui A el DFA de l'apartat anterior. Sigui B un DPDA d'entrada. Hem de comprovar $\mathcal{L}(A) \subseteq \mathcal{L}(B)$ i $\mathcal{L}(A) \supseteq \mathcal{L}(B)$. Sabem que un DFA es pot complementar en un nou DFA reconeixent el llenguatge complementari del primer. Similarment, un DPDA es pot complementar en un nou DPDA reconeixent el llenguatge complementari del primer. També sabem que un DFA i un DPDA es poden intersecar obtenint així un nou DPDA que reconeix el llenguatge intersecció dels dos inicials. Així doncs, $\overline{A} \cap B$ és un DPDA que cumpleix $\mathcal{L}(\overline{A} \cap B) = \overline{\mathcal{L}(A)} \cap \mathcal{L}(B)$, que serà el llenguatge buit si i només si $\mathcal{L}(A) \supseteq \mathcal{L}(B)$. Similarment, $A \cap \overline{B}$ és un DPDA que cumpleix $\mathcal{L}(A \cap \overline{B}) = \mathcal{L}(A) \cap \overline{\mathcal{L}(B)}$, que serà el llenguatge buit si i només si $\mathcal{L}(A) \subseteq \mathcal{L}(B)$. Tot això redueix el nostre problema a decidir si un DPDA donat reconeix el conjunt buit. Però recordem que tot DPDA es pot transformar en una CFG que genera el mateix llenguatge. Per a comprovar si una CFG genera el llenguatge buit n'hi ha prou a calcular els símbols fructífers i verificar si el símbol inicial de la CFG està entre ells.**
3. **Aquest conjunt C no és decidible. Ho justifiquem amb la següent reducció des de K:**

$$x \mapsto p : \left[\begin{array}{l} \texttt{entrada } w \\ \quad \texttt{executar } M_x(x) \\ \quad \texttt{Si } |w|_{ababb} \geqslant 1 \texttt{ llavors acceptar} \\ \quad \texttt{Sino rebutjar} \end{array}\right.$$

La transformació és computable i total. Ademés, preserva la resposta:

Si x és de K, llavors $M_x(x)$ s'atura. Per tant, M_p accepta justament aquells mots que contenen *ababb*, de manera que p és de C.

Si x no és de K, llavors $M_x(x)$ no s'atura. Per tant, M_p rebutja tots els mots, i en particular M_p no reconeix el llenguatge dels mots que contenen *ababb*, de manera que p no és de C.

4. (1,5 punts) Classifiqueu com a decidible, semi-decidible però no decidible, o no semi-decidible el problema de determinar si la funcio computada per una TM donada té imatge infinita. (Recordeu que la imatge $\mathsf{Im}(f)$ d'una funció f és $\{y \mid \exists x \;\; y = f(x)\}$.) Formalment, es tracta de classificar el conjunt

$$\{x \mid \mathsf{Im}(\varphi_x) \text{ és infinita}\}.$$

Resposta: Aquest conjunt, que anomenem C, no és ni semi-decidible. Ho demostrem amb la següent reducció des de $\overline{K}$.

$$x \mapsto p : \left[\begin{array}{l} \texttt{entrada } y \\ \quad \texttt{Si } M_x(x) \texttt{ s'atura en } y \texttt{ o menys passos llavors sortida } 1 \\ \quad \texttt{Sino sortida } y \end{array}\right.$$

La transformació és computable i total. A més a més preserva la resposta:

Si x és de $\overline{K}$ llavors $M_x(x)$ no s'atura. Per tant, la máquina M_p computa la funció identitat, que té imatge infinita.

Si x no és de $\overline{K}$ llavors $M_x(x)$ s'atura en un cert temps d'execució T. Per a entrades majors que T la máquina M_p dona sortida 1. Per tant, φ_p no té imatge infinita.

5. (1,5 punts) Demostreu que és indecidible el problema següent: donades dues gramàtiques, G_1 i G_2, determinar si els llenguatges que generen són complementaris, és a dir que $\mathsf{L}(G_1) = \overline{\mathsf{L}(G_2)}$.

Resposta: Demostrarem que és indecidible amb una reducció des del problema d'universalitat, que hem vist durant el curs que és indecidible:

$$\{G \in \mathtt{CFG} \mid \mathcal{L}(G) = \Sigma^*\}$$

La reducció és la següent:

$$G \;\mapsto\; \langle G, S \to S\rangle$$

La transformació és computable i total. Ademés, preserva la resposta, perque la gramática $S \to S$ genera el llenguatge buit:

$$\mathcal{L}(G) = \Sigma^* \;\Leftrightarrow\; \mathcal{L}(G) = \overline{\emptyset} \;\Leftrightarrow\; \mathcal{L}(G) = \overline{\mathcal{L}(S \to S)}$$

(Examen de Gener-2011)

Per tal d'afavorir als estudiants honestos, en aquest examen, deixar en blanc una pregunta que no se sap respondre puntua més que contestar-la de qualsevol manera per "rascar". Cada pregunta o apartat va acompanyat de la seva valoració quan no es presenta o es deixa en blanc. Però la puntuació en blanc no pot aportar més d'una cinquena part de la nota que s'obtindria sumant punts de la forma tradicional deixant a 0 les respostes en blanc, és a dir:

$\mathtt{Nota\_examen} = \mathtt{Nota\_tradicional} + \mathtt{minim}(\frac{\mathtt{Nota\_tradicional}}{5}, \mathtt{Nota\_en\_blanc})$

1. (1 punt en total) Volem demostrar que un llenguatge L satisfà l'equació $aL = La$ si i només si els mots de L estan formats només pel símbol a. És a dir:

$$aL = La \quad \Leftrightarrow \quad L \subseteq a^*$$

   (a) (0.5 punts, en blanc: 0.1, temps màxim de resolució estimat: 15 minuts) Demostreu la direcció $\Leftarrow$ en detall.

   (b) (0.5 punts, en blanc: 0.1, temps màxim de resolució estimat: indeterminat, doncs **és difícil**) Demostreu la direcció $\Rightarrow$ en detall.

Resposta:

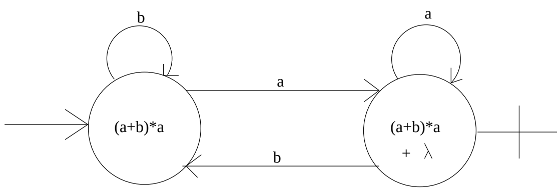
(a) Assumim $L \subseteq a^*$ com a cert. Llavors:

$$aL = \{aw|w \in L\} = \{aa^n|a^n \in L\} = \{a^n a|a^n \in L\} = \{wa|w \in L\} = La$$

(b) **Abans de demostrar la direcció $\Rightarrow$ introduïm notació. Per a un mot w amb algun símbol diferent de a, amb $\texttt{ppos}_{\neq a}(w)$ denotem la primera posició de w on hi ha un símbol diferent de a. També es pot descriure recursivament com $\texttt{ppos}_{\neq a}(aw) = 1 + \texttt{ppos}_{\neq a}(w)$ i $\texttt{ppos}_{\neq a}(cw) = 1$ per a qualsevol símbol c diferent de a. Donat un llenguatge qualsevol L' amb algun mot que contingui algun símbol diferent de a, definim $\texttt{ppos}_{\neq a}(L') = \texttt{min}(\{\texttt{ppos}_{\neq a}(w) \mid w \in L' \wedge w \notin a^*\})$, és a dir, la menor posició tal que apareix un símbol diferent de a en algun mot de L' en aquesta posició.**

Ara demostrem ja la direcció $\Rightarrow$ per contrarecíproc, és a dir, assumim $L \not\subseteq a^*$ i demostrem $aL \neq La$. Sota la nostra assumpció, algun mot de L té algun símbol diferent de a, i el mateix passa pels llenguatges aL i La. Ara bé, $\texttt{ppos}_{\neq a}(aL) = 1 + \texttt{ppos}_{\neq a}(L)$, i $\texttt{ppos}_{\neq a}(La) = \texttt{ppos}_{\neq a}(L)$. Per tant, $\texttt{ppos}_{\neq a}(aL) \neq \texttt{ppos}_{\neq a}(La)$, de manera que $aL \neq La$.

2. (2.5 punts, en blanc: 0.3, temps màxim de resolució estimat: 40 minuts) Diem que un DFA (autòmat finit determinista) té anotacions regulars si cada estat q té com a nom una expressió regular, i aquesta representa el llenguatge que seria reconegut si es comencés sempre l'execució des de q. Per exemple, a continuació tenim un DFA amb anotacions regulars que reconeix el llenguatge dels mots sobre $\{a, b\}$ acabats en a.

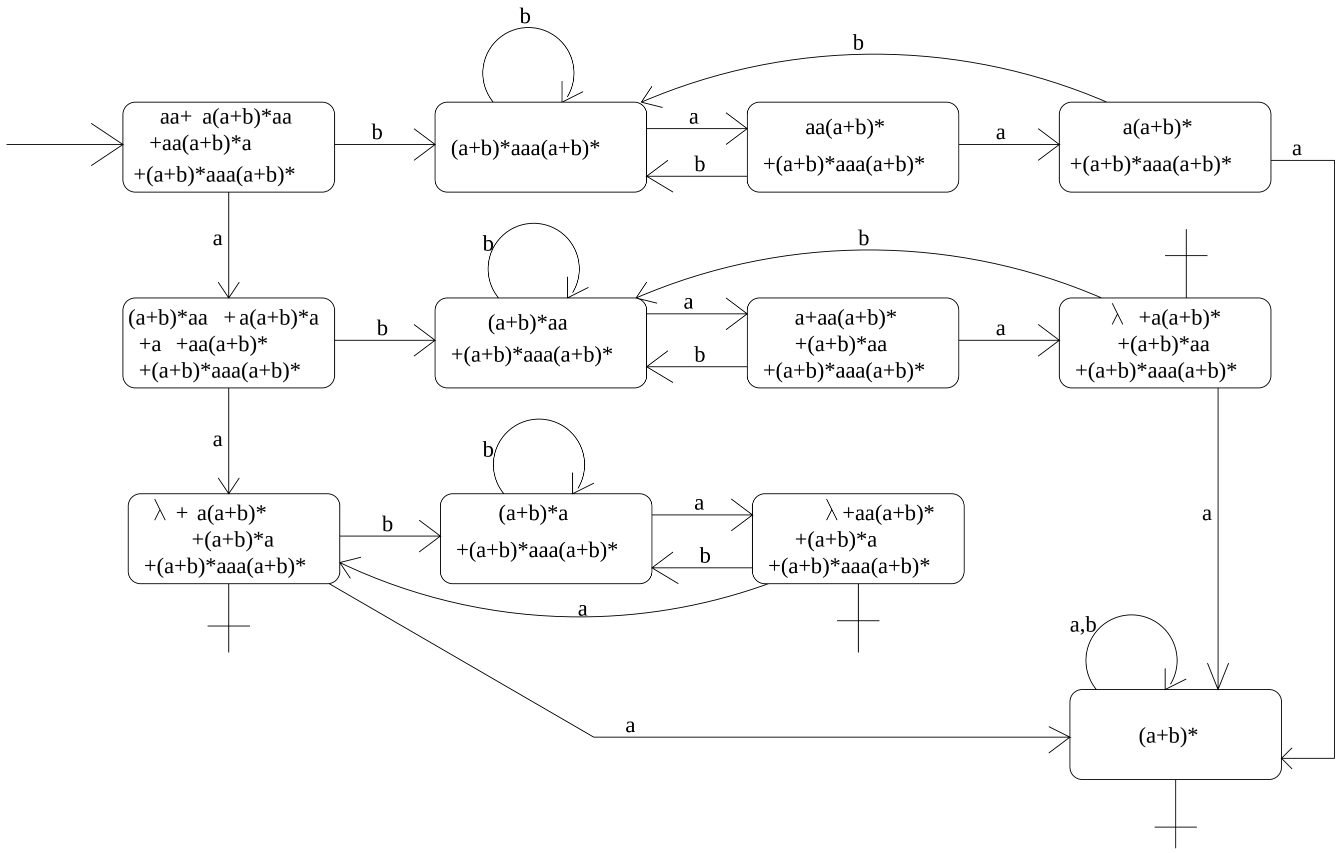


Considereu el llenguatge L_1 dels mots tals que concatenats amb si mateixos contenen al submot aaa. És a dir:

$$L_1 = \{w \in \{a, b\}^* \mid \exists x, y : (ww = xaaay)\}$$

Dibuixeu el DFA mínim A que reconeix L_1 amb anotacions regulars. **Només s'ha de dibuixar el DFA. No cal cap justificació, doncs les anotacions regulars ja serveixen per guiar una justificació. Tampoc cal que demostreu que és mímim, tot i que val la pena que ho comproveu pel vostre compte**.

Resposta:



3. (1.5 punts, en blanc: 0.2, temps màxim de resolució estimat: 25 minuts) Volem obtenir una CFG (gramàtica incontextual) que generi el llenguatge L_2 dels mots palíndroms que contenen 3 ocurrències del submot aaa comptant encavalcaments (recordeu que $|aaaa|_{aaa} = 2$). És a dir, una CFG que generi:

$$L_2 = \{w \in \{a, b\}^* \mid w = w^R \wedge |w|_{aaa} = 3\}$$

Escriviu una CFG que generi L_2. Com a justificació, **únicament cal que descriviu el llenguatge generat per cada variable de la CFG. Feu-ho formalment, amb notació clàssica de conjunts.**

Resposta:

$S \rightarrow bSb \mid abSba \mid aabSbaa \mid aaabXbaaa \mid aaaaa$
$X \rightarrow bXb \mid abXba \mid aabXbaa \mid aaa$

$\mathcal{L}(S) = L_2$
$\mathcal{L}(X) = \{w \in \{a, b\}^* \mid w = w^R \wedge |w|_{aaa} = 1\}$

4. (3 punts en total, temps màxim de resolució estimat: 50 minuts) Justificant cada resposta, classifiqueu com a decidible, semi-decidible però no decidible, o no semi-decidible, els següents conjunts:

   (a) (1 punt, en blanc: 0.15) $C_1 = \{p \mid \mathtt{Dom}(\varphi_p) \text{ és infinit} \wedge \mathtt{Im}(\varphi_p) \text{ és finit}\}$

(b) (1 punt, en blanc: 0.15) $C_2 = \{p \mid \texttt{Dom}(\varphi_p)$ és finit $\wedge$ $\texttt{Im}(\varphi_p)$ és infinit$\}$

(c) (1 punt, en blanc: 0.15) $C_3 = \{p \mid |\texttt{Dom}(\varphi_p) \cup \texttt{Im}(\varphi_p)| \geqslant 2\}$

Nota: Recordeu que $\texttt{Dom}(\varphi_p)$ és el conjunt domini de la funció computada per la màquina codificada per p, és a dir, el conjunt de les entrades per a les quals la màquina M_p dona sortida. Recordeu també que $\texttt{Im}(\varphi_p)$ és el conjunt imatge de la funció computada per la màquina codificada per p, és a dir, el conjunt dels valors que s'obtenen com a sortida de la màquina M_p per a alguna entrada.

Resposta:

(a) Demostrem que C_1 no és ni semi-decidible amb la següent transformació, que ens redueix $\bar{K}$ a C_1:

$$x \mapsto p \left\{ \begin{array}{l} \textbf{Entrada y} \\ \quad \textbf{Si } M_x(x) \textbf{ s'atura en } y \textbf{ passos sortida } y \\ \quad \textbf{Sino sortida } 1 \end{array} \right.$$

És clar que aquesta transformació és computable i total. Veiem que preserva la resposta entre tots dos problemes.

Si x és de $\bar{K}$, llavors $M_x(x)\uparrow$, de manera que φ_p és la funció constant 1, que té domini $\mathbb{N}$ i imatge $\{1\}$, i per tant $p \in C_1$.

Si x no és de $\bar{K}$, llavors $M_x(x)\downarrow$ en un cert nombre de passos T, de manera que $\varphi_p(y) = y$ per a tot $y \geqslant T$. Així doncs, $\texttt{Im}(\varphi_p)$ té infinits elements, i per tant $p \notin C_1$.

Com a resposta alternativa a aquest apartat, tenim que la següent reducció també és correcta:

$$x \mapsto p \left\{ \begin{array}{l} \textbf{Entrada y} \\ \quad \textbf{Si } y \in \dot{2} \textbf{ llavors:} \\ \quad\quad \textbf{Executar } M_x(x) \\ \quad\quad \textbf{sortida } y \\ \quad \textbf{Sino sortida } 0 \end{array} \right.$$

És clar que aquesta transformació és computable i total. Veiem que preserva la resposta entre tots dos problemes.

Si x és de $\bar{K}$, llavors $M_x(x)\uparrow$, de manera que $\texttt{Dom}(\varphi_p) = \dot{2}+1$ i $\texttt{Im}(\varphi_p) = \{0\}$, i per tant $p \in C_1$.

Si x no és de $\bar{K}$, llavors $M_x(x)\downarrow$, de manera que $\texttt{Im}(\varphi_p(y)) = \dot{2}$. Així doncs, $\texttt{Im}(\varphi_p)$ té infinits elements, i per tant $p \notin C_1$.

(b) Qualsevol funció amb domini finit té també imatge finita, doncs la imatge de qualsevol element del domini està unívocament definida. Així doncs, la condició que defineix C_2 no es pot cumplir per cap p, i per tant C_2 és $\emptyset$, que és un conjunt decidible, com ho justifica el fet que la següent màquina de parada segura el reconeix:

```
Entrada y
 Rebutjar
```

(c) Demostrem que C_3 no és decidible amb la següent transformació, que ens redueix K a C_3:

$$x \mapsto p \begin{cases} \textbf{Entrada y} \\ \quad \textbf{Executar } M_x(x) \\ \quad \textbf{sortida } 1 \end{cases}$$

És clar que aquesta transformació és computable i total. Veiem que preserva la resposta entre tots dos problemes.

Si x és de K, llavors el domini de φ_p és $\mathbb{N}$, de manera que $p \in C_3$.

Si x no és de K, llavors $\texttt{Dom}(\varphi_p) = \texttt{Im}(\varphi_p) = \emptyset$, de manera que $p \notin C_3$.

(Aquesta part també es podria haver demostrat utilitzant el Teorema de Rice).

Ara demostrem que C_3 sí és semi-decidible. En primer lloc, notem que C_3 és pot escriure com la unió dels conjunts $C_{31} = \{p \mid |\texttt{Dom}(\varphi_p)| \geqslant 2\}$ i $C_{32} = \{p \mid \exists x, y : (x \neq y \wedge M_p(x) = y)\}$. Donat que els conjunts semi-decidibles són tancats per unió, per acabar n'hi ha prou amb que demostrem que tots dos conjunts són semi-decidibles. Ho fem mostrant programes que els semi-decideixen.

El següent programa semi-decideix C_{31}:

```
Entrada p
   t:=1
   executar indefinidament:
     Per cada x,y en {0,...,t} fer:
       Si x ≠ y i M_p(x) i M_p(y) donen sortida en ⩽ t passos llavors:
         acceptar
     t++
```

Argumentem que efectivament aquesta màquina semi-decideix C_{31}.

Si p és de C_{31}, llavors $|\texttt{Dom}(\varphi_p)| \geqslant 2$, de manera que existeixen dues entrades diferents x, y amb les quals M_p dona sortida. La nostra màquina tard o d'hora considerarà aquestes entrades amb prou nombre de passos, de manera que acabarà acceptant.

Si la màquina accepta p, llavors existeixen dues entrades diferents x, y amb les quals M_p dona sortida, de manera que p és de C_{31}.

El següent programa semi-decideix C_{32}:

```
Entrada p
   t:=1
   executar indefinidament:
     Per cada x,y en {0,...,t} fer:
       Si x ≠ y i M_p(x) dona sortida y en ⩽ t passos llavors:
         acceptar
     t++
```

Argumentem que efectivament aquesta màquina semi-decideix C_{32}.

Si p és de C_{32}, llavors existeixen dos naturals diferents x, y tals que M_p amb entrada x dona sortida y. La nostra màquina tard o d'hora considerarà aquests x, y amb prou nombre de passos, de manera que acabarà acceptant.

Si la màquina accepta p, llavors existeixen dos naturals diferents x, y tals que M_p amb entrada x dona sortida y, de manera que p és de C_{32}.

5. (2 punts en total, temps màxim de resolució estimat: 40 minuts) Durant el curs hem demostrat la no decidibilitat del problema d'universalitat de gramàtiques, que correspon al conjunt de gramàtiques que generen tots els mots sobre l'alfabet. No és difícil veure que aquest problema és indecidible fins i tot si l'alfabet és binari. Denotem com $\texttt{CFG}_{\{a,b\}}$ al conjunt de gramàtiques que tenen per alfabet $\{a,b\}$, i denotem com $\texttt{CFG}_{(a+b)^*}$ al problema d'universalitat restringit al cas en que l'alfabet és $\{a,b\}$.

$$\texttt{CFG}_{(a+b)^*} = \{G \in \texttt{CFG}_{\{a,b\}} \mid \mathcal{L}(G) = (a+b)^*\}$$

(a) (1 punt, en blanc: 0.15) Amb una reducció des de $\texttt{CFG}_{(a+b)^*}$, demostreu la indecidibilitat del conjunt de les gramátiques que generen el llenguatge dels mots sobre $\{a,b\}$ que comencen per aaa (justifiqueu la resposta):

$$\texttt{CFG}_{aaa(a+b)^*} = \{G \in \texttt{CFG}_{\{a,b\}} \mid \mathcal{L}(G) = aaa(a+b)^*\}$$

(b) (1 punt, en blanc: 0.15) Amb una reducció des de $\texttt{CFG}_{(a+b)^*}$, també volem demostrar la indecidibilitat del conjunt de les gramátiques que generen el llenguatge dels mots sobre $\{a,b\}$ que contenen al submot aaa:

$$\texttt{CFG}_{(a+b)^*aaa(a+b)^*} = \{G \in \texttt{CFG}_{\{a,b\}} \mid \mathcal{L}(G) = (a+b)^*aaa(a+b)^*\}$$

Completeu la següent reducció de manera que assoleixi l'objectiu (justifiqueu la resposta):

$$\left[\begin{array}{ccl} X_1 & \to & w_{11}|\dots|w_{1m_1} \\ X_2 & \to & w_{21}|\dots|w_{2m_2} \\ \vdots & \vdots & \vdots \\ X_n & \to & w_{n1}|\dots|w_{nm_n} \end{array}\right] \mapsto \left[\begin{array}{ccl} S & \to & S_1|S_2 \\ S_1 & \to & bS_1'|abS_1'|aabS_1' \\ S_1' & \to & aS_1'|bS_1'|aaaS_1'' \\ S_1'' & \to & aS_1''|bS_1''|\lambda \\ S_2 & \to & \dots ? \dots \\ \vdots & \vdots & \vdots \\ ? & \vdots & ? \\ \vdots & \vdots & \vdots \end{array}\right]$$

Resposta:

(a) La següent transformació redueix $\texttt{CFG}_{(a+b)^*}$ a $\texttt{CFG}_{aaa(a+b)^*}$:

$$\left[\begin{array}{ccl} X_1 & \to & w_{11}|\dots|w_{1m_1} \\ X_2 & \to & w_{21}|\dots|w_{2m_2} \\ \vdots & \vdots & \vdots \\ X_n & \to & w_{n1}|\dots|w_{nm_n} \end{array}\right] \mapsto \left[\begin{array}{ccl} S & \to & aaaX_1 \\ X_1 & \to & w_{11}|\dots|w_{1m_1} \\ X_2 & \to & w_{21}|\dots|w_{2m_2} \\ \vdots & \vdots & \vdots \\ X_n & \to & w_{n1}|\dots|w_{nm_n} \end{array}\right]$$

És clar que aquesta transformació és computable i total. Ademés, des de X_1 es genera $(a+b)^*$ si i només si des de S es genera $aaa(a+b)^*$. Per tant, aquesta transformació preserva la resposta entre tots dos problemes.

(b) Completem la transformació de l'enunciat per a convertir-la en una reducció de $\texttt{CFG}_{(a+b)^*}$ en $\texttt{CFG}_{(a+b)^*aaa(a+b)^*}$:

$$\left[\begin{array}{lcl} X_1 & \to & w_{11}|\dots|w_{1m_1} \\ X_2 & \to & w_{21}|\dots|w_{2m_2} \\ \vdots & \vdots & \vdots \\ X_n & \to & w_{n1}|\dots|w_{nm_n} \end{array}\right] \mapsto \left[\begin{array}{lcl} S & \to & S_1|S_2 \\ S_1 & \to & bS_1'|abS_1'|aabS_1' \\ S_1' & \to & aS_1'|bS_1'|aaaS_1'' \\ S_1'' & \to & aS_1''|bS_1''|\lambda \\ S_2 & \to & aaaX_1 \\ X_1 & \to & w_{11}|\dots|w_{1m_1} \\ X_2 & \to & w_{21}|\dots|w_{2m_2} \\ \vdots & \vdots & \vdots \\ X_n & \to & w_{n1}|\dots|w_{nm_n} \end{array}\right]$$

És clar que aquesta transformació és computable i total. Des de S_1 es genera el llenguatge dels mots sobre $\{a, b\}$ que contenen aaa, però que aaa no és prefix del mot, és a dir, $\{w \in \{a, b\}^* \mid |w|_{aaa} \geqslant 1 \wedge \nexists w' : w = aaaw'\}$, que de fet és $(a + b)^*aaa(a + b)^* - aaa(a + b)^*$. Ademés, des de X_1 es genera $(a + b)^*$ si i només si des de S_2 es genera $aaa(a + b)^*$. Així doncs, des de X_1 es genera $(a + b)^*$ si i només si des de S es genera $(a+b)^*aaa(a+b)^*$. Per tant, aquesta transformació preserva la resposta entre tots dos problemes.

(Examen de Juny-2011)

Per tal d'afavorir als estudiants honestos, en aquest examen, deixar en blanc una pregunta que no se sap respondre puntua més que contestar-la de qualsevol manera per "rascar". Cada pregunta o apartat va acompanyat de la seva valoració quan no es presenta o es deixa en blanc. Però la puntuació en blanc no pot aportar més d'una cinquena part de la nota que s'obtindria sumant punts de la forma tradicional deixant a 0 les respostes en blanc, és a dir:

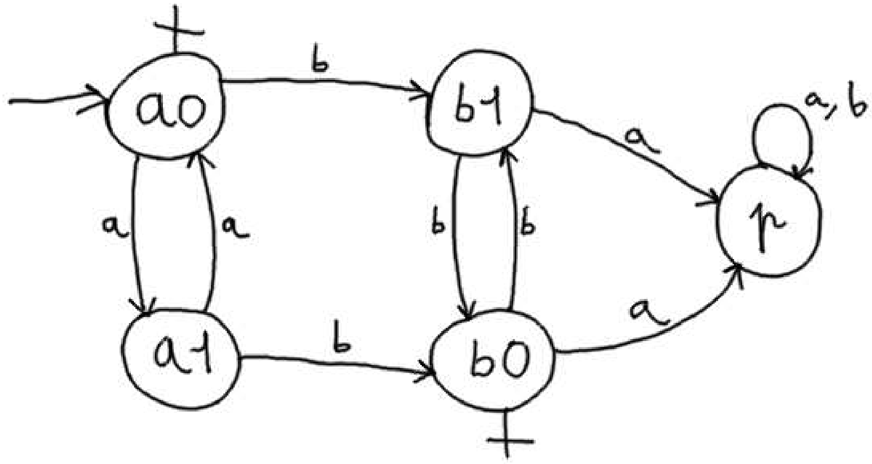
$\texttt{Nota\_examen} = \texttt{Nota\_tradicional} + \texttt{minim}(\frac{\texttt{Nota\_tradicional}}{5}, \texttt{Nota\_en\_blanc})$

1. (2.5 punts en total)
   (a) (1.5 punts, en blanc: 0.2, temps màxim de resolució estimat: 25 minuts) Dibuixeu el DFA mínim que reconeix el conjunt de mots de longitud parell de a^*b^*, és a dir, $\{a^nb^m \mid (n + m) \in \dot{2}\}$. Com a justificació, no cal que expliqueu com obteniu el DFA. Només cal que digueu, per a cada estat, quin és el conjunt de mots que ens porten des de l'estat inicial fins a aquell estat. Mostreu també l'aplicació de l'algorisme de minimització sobre el vostre DFA per tal de demostrar que és mínim.
   (b) (1 punt, en blanc: 0.15, temps màxim de resolució estimat: 20 minuts) Descriviu un algorisme, a grans trets i de forma molt resumida, per tal de decidir si el llenguatge reconegut per un DFA donat inclou a tots els mots de longitud parell sobre $\{a, b\}$; és a dir, que resolgui el següent problema:

   Entrada: Un DFA A amb alfabet $\{a, b\}$.
   Qüestió: $\{w \in \{a, b\}^* \mid |w| \in \dot{2}\} \subseteq \mathcal{L}(A)$?

Resposta:

(a)



Amb $\mathcal{L}(q)$ denoto el llenguatge dels mots que em porten des de l'estat inicial a l'estat q:

$\mathcal{L}(a0) = \{a^n \mid n \in \dot{2}\}$
$\mathcal{L}(a1) = \{a^n \mid n \in \dot{2}+1\}$
$\mathcal{L}(b0) = \{a^n b^m \mid n \geqslant 0, m \geqslant 1, (n+m) \in \dot{2}\}$
$\mathcal{L}(b1) = \{a^n b^m \mid n \geqslant 0, m \geqslant 1, (n+m) \in \dot{2}+1\}$
$\mathcal{L}(p) = (a+b)^* ba (a+b)^*$

Execució de l'algorisme de minimització:

Partició en 0-indistingibles: $\{a0, b0\}\{a1, b1, p\}$.
Partició en 1-indistingibles: $\{a0, b0\}\{a1\}\{b1\}\{p\}$.
Partició en 2-indistingibles: $\{a0\}\{b0\}\{a1\}\{b1\}\{p\}$.
Partició en 3-indistingibles: $\{a0\}\{b0\}\{a1\}\{b1\}\{p\}$.

(b) La condició $\{w \in \{a,b\}^* \mid |w| \in \dot{2}\} \subseteq \mathcal{L}(A)$ és equivalent a $\{w \in \{a,b\}^* \mid |w| \in \dot{2}\} \cap \overline{\mathcal{L}(A)} = \emptyset$. Donat que $\{w \in \{a,b\}^* \mid |w| \in \dot{2}\}$ i $\mathcal{L}(A)$ són regulars, i que els llenguatges regulars són tancats per complementari i intersecció, podem construïr automàticament un DFA B que reconegui $\{w \in \{a,b\}^* \mid |w| \in \dot{2}\} \cap \overline{\mathcal{L}(A)}$. A continuació cal comprovar si el llenguatge reconegut per B és buit o no. Això es pot decidir, per exemple, comprovant si hi ha algun estat acceptador accessible des de l'estat inicial de B.

2. (2 punts en total)

   (a) (1 punt, en blanc: 0.15, temps màxim de resolució estimat: 20 minuts) Dibuixeu una CFG que generi el conjunt de mots $\{a^i b^j c^k \mid j = i + k \ \wedge \ i + j \in \dot{2}\}$. Com a justificació, només cal que digueu, per a cada variable de la gramàtica, quin és el conjunt de mots que genera.

(b) (1 punt, en blanc: 0.15, temps màxim de resolució estimat: 20 minuts) Descriviu un algorisme, a grans trets i de forma molt resumida, per tal de decidir si el llenguatge generat per una CFG donada està inclòs en el llenguatge de tots els mots de longitud parell sobre $\{a, b\}$; és a dir, que resolgui el següent problema:

Entrada: Una CFG G amb alfabet de símbols terminals $\{a, b\}$.

Questió: $\mathcal{L}(G) \subseteq \{w \in \{a, b\}^* \mid |w| \in \dot{2}\}$?

Resposta:

(a) Proposem la següent CFG:

$$\begin{aligned} S &\to XY \\ X &\to aXb \mid \lambda \\ Y &\to bbYcc \mid \lambda \end{aligned}$$

$\mathcal{L}(S) = \{a^i b^j c^k \mid j = i + k \ \wedge \ i + j \in \dot{2}\}$
$\mathcal{L}(X) = \{a^n b^n \mid n \geqslant 0\}$
$\mathcal{L}(Y) = \{b^{2n} c^{2n} \mid n \geqslant 0\}$

(b) La condició $\mathcal{L}(G) \subseteq \{w \in \{a, b\}^* \mid |w| \in \dot{2}\}$ és equivalent a $\mathcal{L}(G) \cap \{w \in \{a, b\}^* \mid |w| \in \dot{2} + 1\} = \emptyset$. Donat que $\mathcal{L}(G)$ és incontextual, $\{w \in \{a, b\}^* \mid |w| \in \dot{2} + 1\}$ és regular, i que la intersecció d'un llenguatge incontextual amb un regular dona lloc a un llenguatge incontextual, podem construïr automàticament una CFG G' que generi $\mathcal{L}(G) \cap \{w \in \{a, b\}^* \mid |w| \in \dot{2} + 1\}$. A continuació cal comprovar si el llenguatge generat per G' és buit. Això es pot decidir, per exemple, calculant els símbols fructífers i veient si el símbol inicial de G' no és fructífer.

3. (2.5 punts en total) Classifiqueu justificadament com a decidibles, semi-decidibles però no decidibles, o no semi-decidibles, els següents conjunts (tot i que se suposa que els programes reben naturals com a entrada, quan calgui es pot assumir que aquestes entrades codifiquen mots sobre $\{a, b\}^*$ de forma biunívoca):

(a) (1.25 punts, en blanc: 0.2, temps màxim de resolució estimat: 20 minuts) Els naturals tals que el domini de la funció implementada pel programa que codifiquen és el conjunt dels mots de mida parell sobre $\{a, b\}$, és a dir:

$$L_1 = \{p \mid \mathtt{Dom}(\varphi_p) = \{w \in \{a, b\}^* \mid |w| \in \dot{2}\}\}$$

(b) (1.25 punts, en blanc: 0.2, temps màxim de resolució estimat: 20 minuts) Els naturals tals que el domini de la funció implementada pel programa que codifiquen no és un subconjunt dels mots de mida parell sobre $\{a, b\}$, és a dir:

$$L_2 = \{p \mid \mathtt{Dom}(\varphi_p) \not\subseteq \{w \in \{a, b\}^* \mid |w| \in \dot{2}\}\}$$

Resposta:

(a) Demostrem que L_1 és no semi-decidible amb la següent reducció des de $\bar{K}$:

$$x \mapsto p = \left\{ \begin{array}{l} \textbf{Entrada } w \\ \quad \textbf{Si } |w| \in \dot{2} \textbf{ llavors sortida 1} \\ \quad \textbf{Executar } M_x(x) \\ \quad \textbf{sortida 1} \end{array} \right.$$

És evident que aquesta transformació és computable i total. Justifiquem a continuació que preserva la resposta entre $\bar{K}$ i L_1.
Si $x \in \bar{K}$ llavors $M_x(x) \uparrow$, de manera que $\texttt{Dom}(\varphi_p) = \{w \in \{a,b\}^* \bigm| |w| \in \dot{2}\}$, i per tant $p \in L_1$.
Si $x \notin \bar{K}$ llavors $M_x(x) \downarrow$, de manera que $\texttt{Dom}(\varphi_p) = \{a,b\}^*$, i per tant $p \notin L_1$.

(b) **Demostrarem que L_2 és semi-decidible però no decidible.**
Comencem demostrant L_2 no és decidible utilitzant el Teorema de Rice. En primer lloc notem que L_2 és un conjunt d'índexos degut a que la propietat que defineix el conjunt es refereix a φ_p. A més a més, L_2 no és el conjunt total ja que el programa $p = \{\texttt{Entrada}\ w\ \texttt{ens penjem}\}$ no és de L_2. Tampoc és el buit, ja que el programa $p = \{\texttt{Entrada}\ w\ \texttt{sortida 1}\}$ és de L_2. Això conclou que L_2 no és decidible.
Demostrem que L_2 sí és semi-decidible utilitzant el teorema de projecció. En primer lloc notem que L_2 es pot descriure també de la següent manera:

$$L_2 = \{p \mid \exists w, t : |w| \in \dot{2} + 1 \wedge M_p(w) \texttt{ dona sortida en } t \texttt{ passos}\}$$

En segon lloc notem que la condició de l'existencial és decidible per a w, t fixats. Això conclou que L_2 sí és semi-decidible.

4. (3 punts en total) Considerem els següents problemes/conjunts sobre gramàtiques.

`Universalitat` (decidir si una gramàtica incontextual amb alfabet $\{a,b\}$ genera tots els mots sobre $\{a,b\}$):

$$U = \{G \in \texttt{CFG}_{\{a,b\}} \mid \mathcal{L}(G) = \{a,b\}^*\}$$

`Universalitat-parell` (decidir si una gramàtica incontextual amb alfabet $\{a,b\}$ genera exactament tots els mots sobre $\{a,b\}$ de mida parell):

$$U_{\texttt{parell}} = \{G \in \texttt{CFG}_{\{a,b\}} \mid \mathcal{L}(G) = \{w \in \{a,b\}^* \bigm| |w| \in \dot{2}\}\}$$

`Universalitat-senar` (decidir si una gramàtica incontextual amb alfabet $\{a,b\}$ genera exactament tots els mots sobre $\{a,b\}$ de mida senar):

$$U_{\texttt{senar}} = \{G \in \texttt{CFG}_{\{a,b\}} \mid \mathcal{L}(G) = \{w \in \{a,b\}^* \bigm| |w| \in \dot{2} + 1\}\}$$

Durant el curs hem vist que U és indecidible. Volem demostrar que $U_{\texttt{parell}}$ i $U_{\texttt{senar}}$ també ho són mitjançant les reduccions $U \leqslant U_{\texttt{parell}}$ i $U_{\texttt{parell}} \leqslant U_{\texttt{senar}}$.

(a) (1.5 punts, en blanc: 0.2, temps màxim de resolució estimat: 20 minuts) Doneu una reducció $U_{\texttt{parell}} \leqslant U_{\texttt{senar}}$ justificant la resposta.
(b) (1.5 punts, en blanc: 0.2, temps màxim de resolució estimat: 35 minuts) Doneu una reducció $U \leqslant U_{\texttt{parell}}$ justificant la resposta.

(a) **Proposem la següent reducció, on S' és una variable nova:**

$$G = \langle V, \Sigma, \delta, S \rangle \ \mapsto\ G' = \langle V \cup \{S'\}, \Sigma, \delta \cup \{S' \to aS|bS\}, S' \rangle$$

És evident que la transformació és computable i total, i que G genera tots els mots de mida parell sobre $\{a,b\}$ si i només si G' genera tots els mots de mida senar sobre $\{a,b\}$.

(b) Proposem la següent reducció, on A, B són variables noves i σ el morfisme que transforma a en A, b en B, i és la identitat per a qualsevol altre símbol:

$$G = \langle V, \Sigma, \delta, S\rangle \;\mapsto\; G' = \langle V \cup \{A, B\}, \Sigma, \sigma(\delta) \cup \{A \to aa|ab,\ B \to ba|bb\}, S\rangle$$

És evident que la transformació és computable i total, i que G genera tots els mots sobre $\{a, b\}$ si i només si G' genera tots els mots de mida parell sobre $\{a, b\}$.

Com a resposta alternativa, la següent reducció també funciona, on S', X són variables noves i σ el morfisme que transforma a en aa, b en bb, i és la identitat per a qualsevol altre símbol:
$G = \langle V, \Sigma, \delta, S\rangle \;\mapsto$
$G' = \langle V \cup \{S', X\}, \Sigma, \sigma(\delta) \cup \{S' \to S|XbaX|XabX, X \to abX|baX|aaX|bbX|\lambda\}, S'\rangle$

(Examen de Gener-2012)

Per tal d'afavorir als estudiants honestos, en aquest examen, deixar en blanc una pregunta que no se sap respondre puntua més que contestar-la de qualsevol manera per "rascar". Cada pregunta o apartat va acompanyat de la seva valoració quan no es presenta o es deixa en blanc. Però la puntuació en blanc no pot aportar més d'una cinquena part de la nota que s'obtindria sumant punts de la forma tradicional deixant a 0 les respostes en blanc, és a dir:

$\texttt{Nota\_examen} = \texttt{Nota\_tradicional} + \texttt{minim}(\frac{\texttt{Nota\_tradicional}}{5}, \texttt{Nota\_en\_blanc})$

1. (4 punts en total) Definim un llenguatge L_k en funció d'un paràmetre k així: un mot és de L_k si té mida major o igual a k, i el seu prefix de mida k coincideix amb el seu sufix de mida k. Formalment:

$$L_k = \{w \in \{a, b\}^* \mid \exists u, x, y : (|u| = k \wedge w = ux \wedge w = yu)\}$$

   Per exemple, fixeu-vos en que aba pertany a L_0, L_1, L_3 però no pertany a L_2 ni L_4.

   Resoleu les següents qüestions:

   (a) (1.5 punts, en blanc: 0.15, temps de resolució estimat: 20 minuts) Dibuixeu els DFA's mínims dels llenguatges L_0, L_1, L_2. Només cal que dibuixeu els DFA's, no cal cap justificació ni demostrar que són mínims.

   (b) (1.25 punts, en blanc: 0.15, temps de resolució estimat: 20 minuts) Descriviu raonadament una CFG no ambigua que generi $L = L_2 \cap \{w \in \{a, b\}^* \mid w = w^R\}$. Comenceu descrivint la forma dels mots del llenguatge L.

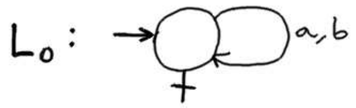
   (c) (1.25 punts, en blanc: 0.15, temps de resolució estimat: 20 minuts) Sigui L_{ab} el llenguatge dels mots ben parentitzats on a representa el parèntesi d'obrir, i b representa el parèntesi de tancar, és a dir:

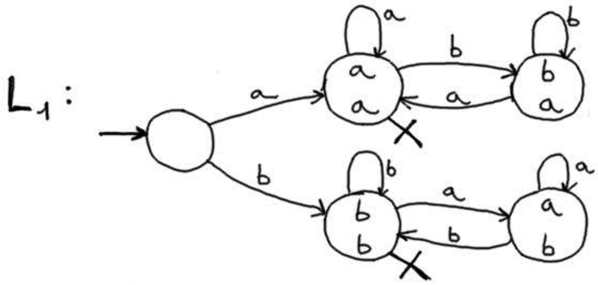
$$L_{ab} = \{w \in \{a, b\}^* \bigm| |w|_a = |w|_b \wedge \forall x, y : (w = xy \Rightarrow |x|_a \geqslant |x|_b)\}$$

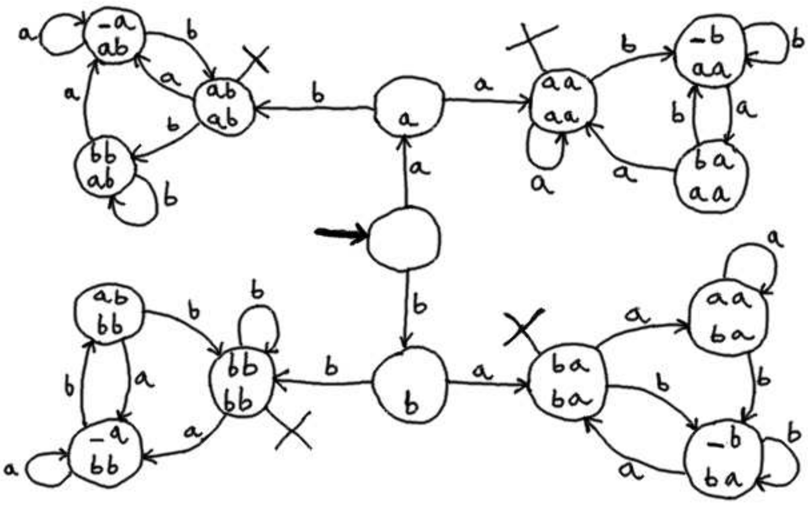
   Descriviu raonadament una CFG no ambigua que generi $L' = L_2 \cap L_{ab}$. Comenceu descrivint la forma dels mots del llenguatge L'.

Resposta:

(a)







(b) Tot mot igual al seu revessat i de mida almenys 2 comença i acaba amb el mateix símbol, i per tant o bé és de la forma ava, o bé és de la forma bvb. Tot mot de L_2 de la forma ava o bvb compleix que el primer símbol coincideix amb el penúltim símbol, i que el segon símbol coincideix amb l'últim símbol. Per tant, tot mot de L ha de ser o bé aa o bé bb, o bé aaa, o bé bbb, o bé de la forma $aavaa$, o bé de la forma $bbvbb$. A més a més, tot mot de la forma $aavaa$ o $bbvbb$ igual al seu revessat compleix que v és igual al seu revessat. Per tant, L es pot descriure com:

$$L = \{aa, bb, aaa, bbb\} \cup \{aavaa \in \{a,b\}^* \mid v = v^R\} \cup \{bbvbb \in \{a,b\}^* \mid v = v^R\}$$

Així doncs, una CFG que genera L és:

$S \rightarrow aa \mid bb \mid aaa \mid bbb \mid aaXaa \mid bbXbb$
$X \rightarrow aXa \mid bXb \mid a \mid b \mid \lambda$

(c) Tot mot ben parentitzat i diferent de λ comença per a i acaba per b. Tot mot de L_2 que comença per a i acaba per b és o bé ab, o bé de la forma $abuab$, ja que el primer

símbol ha de coincidir amb el penúltim, i el segon símbol ha de coincidir amb l'últim. Tot mot de la forma $abuab$ que sigui ben parentitzat ha de complir que el submot u també ha de ser ben parentitzat. Això conclou que els mots de L' són o bé ab, o bé els de la forma $abuab$, on u és un mot ben parentitzat. Per tant, una CFG que genera L' és:

$$\begin{aligned} S &\rightarrow abXab \mid ab \\ X &\rightarrow aXbX \mid \lambda \end{aligned}$$

2. (2.5 punts en total) Classifiqueu raonadament els següents problemes com a decidibles, semi-decidibles però no decidibles, o no semi-decidibles.

   (a) (1.25 punts, en blanc: 0.15, temps de resolució estimat: 20 minuts) El conjunt de naturals tals que el llenguatge reconegut per la màquina que codifiquen inclou $\{1, 13, 18\}$:

   $$C_1 = \{p \mid \{1, 13, 18\} \subseteq \mathcal{L}_p\}$$

   (b) (1.25 punts, en blanc: 0.15, temps de resolució estimat: 20 minuts) El conjunt de naturals tals que el llenguatge reconegut per la màquina que codifiquen inclou $\{1, 13, 18\}$ i té cardinal infinit:

   $$C_2 = \{p \mid \{1, 13, 18\} \subseteq \mathcal{L}_p \ \wedge \ |\mathcal{L}_p| = \infty\}$$

Resposta:

(a) Demostrem que C_1 no és decidible amb la següent transformació, que ens redueix K a C_1:

$$x \mapsto p = \left\{ \begin{array}{l} \textbf{Entrada y} \\ \quad \textbf{Executar } M_x(x) \\ \quad \textbf{Acceptar} \end{array} \right.$$

És clar que aquesta transformació és computable i total. Per demostrar que és una reducció correcta, falta veure que preserva la resposta entre tots dos problemes.

Si x és de K, llavors $M_x(x)\downarrow$, de manera que M_p accepta amb tota entrada, i en particular accepta amb entrades $1, 13, 18$, i per tant es compleix $\{1, 13, 18\} \subseteq \mathcal{L}_p$, així que $p \in C_1$.

Si x no és de K, llavors $M_x(x)\uparrow$, de manera que M_p no accepta cap entrada, i en particular no accepta cap de les entrades $1, 13, 18$, i per tant $\mathcal{L}_p$ no inclou a $\{1, 13, 18\}$, així que $p \notin C_1$.

Ara, demostrem que C_1 sí és semi-decidible mostrant un programa que el semi-decideix:

```
Entrada p
   Si Mp(1) accepta i Mp(13) accepta i Mp(18) accepta llavors Acceptar
   Sino Rebutjar
```

És evident que el programa anterior reconeix C_1. Notem que aquest programa es podria quedar penjat amb entrades que no són de C_1, ja que la comprovació de les condicions sobre $M_p(1), M_p(13), M_p(18)$ pot no acabar, però no importa perquè estem semi-decidint C_1.

(b) **Demostrem que C_2 no és ni semi-decidible amb la següent transformació, que ens redueix $\bar{K}$ a C_2:**

$$x \mapsto p = \begin{cases} \textbf{Entrada y} \\ \quad \textbf{Si } y = 1 \textbf{ o } y = 13 \textbf{ o } y = 18 \textbf{ llavors acceptar} \\ \quad \textbf{Sino Si } M_x(x) \textbf{ s'atura en } y \textbf{ passos llavors Rebutjar} \\ \quad \textbf{Sino Acceptar} \end{cases}$$

És clar que aquesta transformació és computable i total. Per demostrar que és una reducció correcta, falta veure que preserva la resposta entre tots dos problemes.

Si x és de $\bar{K}$, llavors $M_x(x)\uparrow$, de manera que M_p accepta amb tota entrada, tant si aquesta entrada és de $\{1, 13, 18\}$ com si no, i per tant $\mathcal{L}_p$ inclou a $\{1, 13, 18\}$ i té cardinal infinit, així que $p \in C_2$.

Si x no és de $\bar{K}$, llavors $M_x(x)\downarrow$ en un cert nombre de passos T, de manera que $\mathcal{L}_p = \{1, 13, 18\} \cup \{0, 1, 2, \ldots, T-1\}$, i per tant el cardinal de $\mathcal{L}_p$ no és infinit, així que $p \notin C_2$.

3. (3.5 punts en total) Considerem els següents problemes/conjunts, on s'assumeix que ni u ni v ni cap costat de regla de R és el mot buit (de fet s'assumeix que la pròpia codificació de l'entrada exclou aquests casos):

`Accessibilitat` (decidir si un mot u pot accedir a un mot v aplicant regles de reescriptura d'un conjunt R):

$$A = \{\langle u, v, R\rangle \mid u \to_R^* v\}$$

`Accessibilitat-irreversible` (decidir si un mot u pot accedir a un mot v aplicant regles de reescriptura d'un conjunt R, però en canvi que v no pot accedir a u aplicant R):

$$A_{\texttt{irrev}} = \{\langle u, v, R\rangle \mid u \to_R^* v \ \wedge \ v \not\to_R^* u\}$$

`Accessibilitat-reversible` (decidir si un mot u pot accedir a un mot v aplicant regles de reescriptura d'un conjunt R, i també que v pot accedir a u aplicant R):

$$A_{\texttt{rev}} = \{\langle u, v, R\rangle \mid u \to_R^* v \ \wedge \ v \to_R^* u\}$$

Durant el curs hem vist que A és indecidible. Volem demostrar que $A_{\texttt{irrev}}$ i $A_{\texttt{rev}}$ també són indecidibles.

(a) (1 punt, en blanc: 0.15, temps de resolució estimat: 20 minuts) Doneu una reducció $A \leqslant A_{\texttt{irrev}}$ justificant la resposta.

(b) (0.75 punts, en blanc: 0.1, temps de resolució estimat: 20 minuts) Justifiqueu que el següent intent de reducció per justificar $A \leqslant A_{\texttt{rev}}$ és incorrecte:

$$\langle u, v, R\rangle \ \mapsto \ \langle u, v, R \cup \{v \to u\}\rangle$$

Nota: podeu donar un contraexemple, que consisteix en una entrada $\langle u, v, R\rangle$ de A tal que no preserva la resposta, és a dir que no es compleix $(\langle u, v, R\rangle \in A \Leftrightarrow \langle u, v, R \cup \{v \to u\}\rangle \in A_{\texttt{rev}})$.

(c) (0.5 punts, en blanc: 0.1, temps de resolució estimat: 10 minuts) Justifiqueu que el següent intent de reducció per justificar $A \leqslant A_{\texttt{rev}}$ és incorrecte:

$$\langle u, v, R\rangle \ \mapsto \ \begin{cases} \langle a, b, \{a \to b, b \to a\}\rangle & , \text{si } u \to_R^* v \\ \langle a, b, \{a \to b\}\rangle & , \text{si } u \not\to_R^* v \end{cases}$$

(d) (1.25 punts, en blanc: 0.15, temps de resolució estimat: 30 minuts) Doneu una reducció $A \leqslant A_{\texttt{rev}}$ justificant la resposta.

Resposta:

(a) Proposem com a reducció la següent transformació:

$$\langle u, v, R\rangle \mapsto \langle \#, v, R \cup \{\# \to u\}\rangle \qquad \textbf{, on } \# \textbf{ és un símbol nou}$$

És evident que la transformació és computable i total. En primer lloc, veiem que es preserva la resposta afirmativa suposant $u \to_R^* v$. En tal cas es compleix $\# \to_{\# \to u} u \to_R^* v$, i també $v \not\to_{R\cup\{\#\to u\}}^* \#$, perquè v no conté $\#$, i $\#$ no apareix a la part dreta de cap regla. En segon lloc, veiem que es preserva la resposta negativa suposant $\# \to_{R\cup\{\#\to u\}}^* v$ i $v \not\to_{R\cup\{\#\to u\}}^* \#$. Donat que R no conté cap regla amb $\#$, la derivació $\# \to_{R\cup\{\#\to u\}}^* v$ és de la forma $\# \to_{\#\to u} u \to_{R\cup\{\#\to u\}}^* v$. Pel mateix motiu i donat que u no conté $\#$, la derivació $u \to_{R\cup\{\#\to u\}}^* v$ no fa servir la regla $\# \to u$, i per tant és de la forma $u \to_R^* v$.

(b) Considerem l'entrada $\langle a, aa, \{a \to aaa\}\rangle$. Aquesta té resposta negativa a A, ja que no es pot arribar des de a a aa utilitzant la regla $a \to aaa$. La imatge d'aquesta entrada per la transformació de l'enunciat és $\langle a, aa, \{a \to aaa, aa \to a\}\rangle$. Les derivacions $a \to_{a\to aaa} aaa \to_{aa\to a} aa$ i $aa \to_{aa\to a} a$ mostren que aquesta imatge té resposta afirmativa a $A_{\texttt{rev}}$. Per tant, la transformació de l'enunciat no és una reducció correcta.

(c) La transformació de l'enunciat no és una reducció correcta perquè no és computable. Això es veu fàcilment suposant que una certa màquina M la computa, i llavors mostrant la següent màquina que decideix A, cosa que no és possible perquè sabem que A és indecidible:

entrada $\langle u, v, R\rangle$
Si $M(\langle u, v, R\rangle) = \langle a, b, \{a \to b,\ b \to a\}\rangle$ llavors acceptar
sino rebutjar

(d) Proposem com a reducció la següent transformació:

$$\langle u, v, R\rangle \mapsto \langle \#u\#, \#v\#, R' = R \cup \{\#v\# \to \#u\#\}\rangle \qquad \textbf{, on } \# \textbf{ és un símbol nou}$$

És evident que la transformació és computable i total. En primer lloc, veiem que es preserva la resposta afirmativa suposant $u \to_R^* v$. En tal cas es compleix $\#u\# \to_R^* \#v\#$ i $\#v\# \to_{\#v\#\to\#u\#} \#u\#$. Per tant, també es compleix $\#u\# \to_{R'}^* \#v\#$ i $\#v\# \to_{R'}^* \#u\#$. En segon lloc, veiem que es preserva la resposta negativa suposant $\#u\# \to_{R'}^* \#v\#$ i $\#v\# \to_{R'}^* \#u\#$. Per la forma de les regles de R', qualsevol derivació des de $\#u\#$ utilitzant R' arriba a un mot de la forma $\#w\#$, on w no conté el símbol $\#$. A més a més, en la derivació de mida mínima utilitzant R' que ens porta de $\#u\#$ a $\#v\#$, la regla $\#v\# \to \#u\#$ no s'ha fet servir en cap moment, perquè en tal cas, el mot $\#v\#$ hauria estat accedit abans, i per tant aquesta derivació no seria mínima. Així doncs, en la derivació de mida mínima de $\#u\#$ a $\#v\#$ només s'utilitzen regles de R, les quals no contenen al símbol $\#$, i per tant existeix una derivació $u \to_R^* v$.

(Examen de Gener-2012 (GRAU))

Per tal d'afavorir als estudiants honestos, en aquest examen, deixar en blanc una pregunta que no se sap respondre puntua més que contestar-la de qualsevol manera per "rascar". Cada pregunta o apartat va acompanyat de la seva valoració quan no es presenta o es deixa en blanc. Però la puntuació en blanc no pot aportar més d'una cinquena part de la nota que s'obtindria sumant punts de la forma tradicional deixant a 0 les respostes en blanc, és a dir:

$\texttt{Nota\_examen} = \texttt{Nota\_tradicional} + \texttt{minim}(\frac{\texttt{Nota\_tradicional}}{5}, \texttt{Nota\_en\_blanc})$

CONCEPTE PRELIMINAR:

Donat un llenguatge L, *tallar* L dona lloc un nou llenguatge, que denotem $T(L)$, que s'obté agafant cada mot de L de longitud major o igual a 1 i esborrant el seu primer símbol. Formalment:

$$T(L) = \{v \mid \exists u : (|u| = 1 \ \wedge \ uv \in L)\}$$

Per exemple, $T(\{a, baa, abba\}) = \{\lambda, aa, bba\}$, $T(\{\lambda, ab, ba, aab, bab\}) = \{b, a, ab\}$, $T(a^*) = a^*$ $T(\{\lambda\}) = \emptyset$ i $T(\emptyset) = \emptyset$.

PREGUNTES:

1. (a) (0.5 punts, en blanc 0.1, dificultat estimada: baixa, Temps de resolució estimat: 10 minuts) Justifiqueu amb contraexemples que no és cert que les següents igualtats es cumpleixin per a llenguatges L_1, L_2 qualsevols.
   i. $T(L_1 \cap L_2) = T(L_1) \cap T(L_2)$.
   ii. $T(L_1 \cdot L_2) = T(L_1) \cdot L_2$.

   (b) (0.5 punts, en blanc 0.1, dificultat estimada: baixa, Temps de resolució estimat: 10 minuts) Sigui L el conjunt de mots sobre $\{a, b\}$ amb un nombre d'a's múltiple de 3 més 1. Formalment: $L = \{w \in \{a, b\}^* \mid |w|_a \in \dot{3} + 1\}$. Formalitzeu el llenguatge $T(L)$ amb notació clàssica de conjunts, sense més explicació. A més a més, dibuixeu el DFA mínim que reconeix $T(L)$. No cal cap explicació sobre l'autòmat ni justificar que és mínim.

   (c) (0.5 punts, en blanc 0.1, dificultat estimada: baixa, Temps de resolució estimat: 10 minuts) Per al DFA del llenguatge $T(L)$ de l'apartat anterior, mostreu com s'hi aplica el lema d'Arden per a obtenir una expressió regular que representi $T(L)$.

   (d) (0.5 punts, en blanc 0.1, dificultat estimada: baixa, Temps de resolució estimat: 10 minuts) Sigui $L = \{a^n b^n \mid n \geqslant 0\}$. Formalitzeu el llenguatge $T(L)$ amb notació clàssica de conjunts, sense més explicació. A més a més, escriviu una CFG no ambigua que genera $T(L)$. No cal cap explicació sobre la gramàtica.

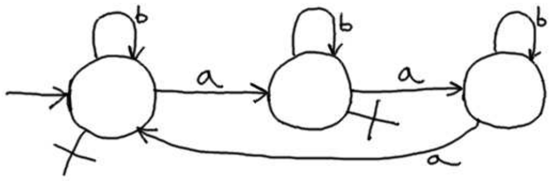
   (e) (0.5 punts, en blanc 0.1, dificultat estimada: baixa, Temps de resolució estimat: 10 minuts) Dibuixeu un autòmat amb pila determinista que reconegui el llenguatge $T(L)$ de l'apartat anterior. No cal cap explicació.

Resposta:

(a) **i.**
$T(\{ab\} \cap \{bb\}) = T(\emptyset) = \emptyset$
$T(\{ab\}) \cap T(\{bb\}) = \{b\} \cap \{b\} = \{b\}$

ii.
$T(\{\lambda\} \cdot \{a\}) = T(\{a\}) = \{\lambda\}$
$T(\{\lambda\}) \cdot \{a\} = \emptyset \cdot \{a\} = \emptyset$

(b) $T(L) = \{w \in \{a,b\}^* \mid |w|_a \in \dot{3} \vee |w|_a \in \dot{3}+1\}$



(c)

$$X_0 = aX_1 + bX_0 + \Lambda$$
$$X_1 = aX_2 + bX_1 + \Lambda$$
$$X_2 = aX_0 + bX_2$$

$$X_2 = b^*aX_0$$
$$X_1 = b^*(aX_2 + \Lambda) = b^*ab^*aX_0 + b^*$$
$$X_0 = a(b^*ab^*aX_0 + b^*) + bX_0 + \Lambda = (ab^*ab^*a + b)X_0 + ab^* + \Lambda$$
$$X_0 = (ab^*ab^*a + b)^*(ab^* + \Lambda)$$

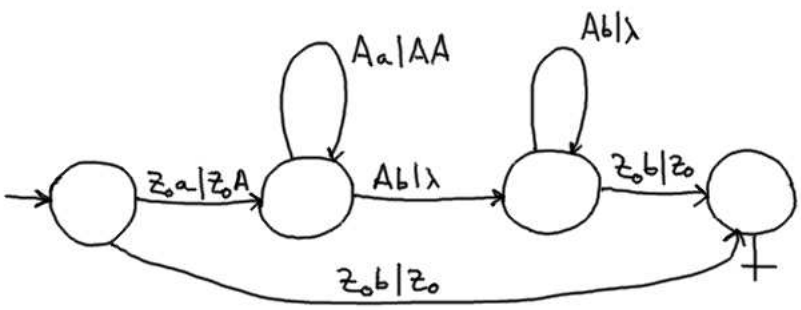
Per tant, $(ab^*ab^*a + b)^*(ab^* + \Lambda)$ **representa** $T(L)$**.**

(d)

$T(L) = \{a^n b^{n+1} \mid n \geqslant 0\}$

$S \rightarrow aSb \mid b$

(e) Una opció seria aplicar el mètode de transformació de CFG's en PDA's sobre la gramàtica de l'apartat anterior. El problema és que aquest mètode dona lloc a un PDA indeterminista, i l'enunciat demana que sigui determinista. Per això construïm directament un PDA determinista:



2. (a) (0.75 punts, en blanc 0.1, dificultat estimada: baixa, Temps de resolució estimat: 15 minuts) Sigui L' un subconjunt no decidible qualsevol de $\{a,b\}^*$, que sabem que existeix com hem vist al llarg del curs. Demostreu que el següent llenguatge no és decidible amb una reducció des d'L':

$$(a \cdot L') \cup (b \cdot \overline{L'})$$

(b) (0.75 punts, en blanc 0.1, dificultat estimada: baixa, Temps de resolució estimat: 15 minuts) Justifiqueu que no és cert que les següents implicacions es cumpleixen per a un llenguatge L qualsevol:

$$\begin{array}{lcl} T(L) \text{ decidible} & \Rightarrow & L \text{ decidible} \\ T(L) \text{ regular} & \Rightarrow & L \text{ regular} \\ T(L) \text{ incontextual} & \Rightarrow & L \text{ incontextual} \end{array}$$

Pista: Feu servir que l'enunciat de l'apartat anterior és cert, tant si l'heu demostrat com si no.

(c) (1 punt, en blanc 0.15, dificultat estimada: mitja, Temps de resolució estimat: 15 minuts) Demostreu que la implicació (L decidible $\Rightarrow T(L)$ decidible) és certa per a qualsevol llenguatge L.

(d) (1 punt, en blanc 0.15, dificultat estimada: alta, Temps de resolució estimat: 20 minuts) Demostreu que la implicació (L regular $\Rightarrow T(L)$ regular) és certa per a qualsevol llenguatge L.

Pista: Podeu explicar com transformar un DFA que reconeix L en un autòmat (no necessàriament determinista) que reconegui $T(L)$.

(e) (1 punt, en blanc 0.15, dificultat estimada: alta, Temps de resolució estimat: 20 minuts) Demostreu que la implicació (L incontextual $\Rightarrow T(L)$ incontextual) és certa per a qualsevol llenguatge L.

Pista: Podeu explicar com transformar una gramàtica en forma normal de Chomsky que reconeix L per a que passi a reconeixer $T(L)$. Alternativament, podeu donar una idea breu i clara de com transformar un autòmat amb pila que reconeix L per a que passi a reconeixer $T(L)$.

Resposta:

(a) Proposem la transformació $w \mapsto aw$ com a reducció. És evident que és computable i total, i a més a més, es cumpleix $w \in L' \Leftrightarrow aw \in ((a \cdot L') \cup (b \cdot \overline{L'}))$.

(b) Considerem el llenguatge $L = ((a \cdot L') \cup (b \cdot \overline{L'}))$ de l'apartat anterior, on s'estableix que aquest llenguatge no és decidible. Per tant, aquest llenguatge tampoc és regular ni incontextual. En canvi, $T(L) = L' \cup \overline{L'} = \Sigma^*$ sí és regular, i per tant també incontextual i decidible. Així doncs, aquest L és un contraexemple a les tres implicacions de l'enunciat.

(c) Sigui L un llenguatge decidible. Llavors existeix una màquina M_L que decideix L. En conseqüència, la següent màquina decideix $T(L)$:

Entrada v
  Per cada $u \in \Sigma$ **fer:**
    Si $M_L(uv)$ **accepta llavors Acceptar**
  Rebutjar

Notem que la màquina para amb tota entrada perque el bucle considera un nombre finit de casos i M_L para amb tota entrada. A més a més, és clar que reconeix $T(L) = \{v \mid \exists u : (|u| = 1 \ \wedge \ uv \in L)\}$.

(d) Sigui L un llenguatge regular qualsevol. Sigui $A = \langle Q, \Sigma, \delta, q_0, F\rangle$ un DFA que reconeix L. Llavors, podem definir un NFA B que reconeix $T(L)$ com $B = \langle Q, \Sigma, \delta, I, F\rangle$, on I és el conjunt d'estats inicials de B, que es defineix com el conjunt d'estats d'A als quals s'hi pot arribar llegint un mot de longitud 1, és a dir, $I = \{q \in Q \mid \exists u : (|u| = 1 \wedge q_0 u = q)\}$. És clar que B reconeix $T(L)$, doncs $v \in \mathcal{L}(B) \Leftrightarrow \exists q \in I : qv \in F \Leftrightarrow \exists u : (|u| = 1 \wedge q_0 uv \in F) \Leftrightarrow \exists u : (|u| = 1 \wedge uv \in L) \Leftrightarrow v \in T(L)$.

(e) **Sigui L un llenguatge incontextual qualsevol. Sigui $G = \langle V, \Sigma, \delta, S\rangle$ una CFG en forma normal de Chomsky que genera L. Definim una nova gramàtica G' que genera $T(L)$ com segueix. Primer la descrivim intuitivament. Si les variables de G són $S, X, Y, Z, \ldots$, llavors hi afegim noves variables $S', X', Y', Z', \ldots$. Per a cada variable X de G, la idea és que la corresponent variable X' generi el llenguatge $T(\mathcal{L}(G, X))$. Per això, per a cada regla $X \to YZ$, afegirem una nova regla $X' \to Y'Z$, i per a cada regla $X \to c$ afegirem una nova regla $X' \to \lambda$. En cas que aparegués la regla $S \to \lambda$, aquesta s'hauria d'eliminar. Per inducció sobre el nombre de passos de reemplaçament es pot demostrar $\mathcal{L}(G', X') = T(\mathcal{L}(G, X))$. La nova variable S' serà el nou símbol inicial. La definició formal de G' és $G' = \langle V \uplus \{X' \mid X \in V\}, \Sigma, (\delta - (\{S \to \lambda\})) \uplus \{X' \to Y'Z \mid (X \to YZ) \in \delta\} \uplus \{X' \to \lambda \mid \exists c \in \Sigma : (X \to c) \in \delta\}, S'\rangle$.**

Resposta alternativa d'aquest apartat: Sigui L un llenguatge incontextual qualsevol. Sigui P un PDA que reconeix L. Podem construïr un PDA P', no necessàriament determinista, que reconeix $T(L)$ com segueix. P', abans de llegir el mot d'entrada v, simularà l'execució de P amb entrada un mot u de mida 1, generat no determinísticament. Un cop fet això, seguirà simulant P des de la configuració a on hagi arribat, però ara sí, llegint l'entrada v. D'aquesta manera, P' accepta v si i només si existeix un mot u de mida 1 tal que P accepta uv.

3. Per als dos primers subapartats d'aquest problema, recordeu que, tot i que se suposa que els programes reben naturals com a entrada, quan calgui es pot assumir que aquestes entrades codifiquen mots sobre $\{a, b\}^*$ de forma biunívoca.

   (a) (1 punt, en blanc 0.15, dificultat estimada: mitja, Temps de resolució estimat: 15 minuts) Classifiqueu raonadament com a decidible, semi-decidible però no decidible, o no semi-decidible, el següent conjunt:

   $$L = \{\langle p, q\rangle \mid \mathcal{L}_p = T(\mathcal{L}_q)\}$$

   (b) (1 punt, en blanc 0.15, dificultat estimada: mitja, Temps de resolució estimat: 15 minuts) Classifiqueu raonadament com a decidible, semi-decidible però no decidible, o no semi-decidible, el següent conjunt:

   $$L = \{p \mid \exists q : \mathcal{L}_p = T(\mathcal{L}_q)\}$$

   (c) (1 punt, en blanc 0.15, dificultat estimada: mitja, Temps de resolució estimat: 15 minuts) Durant el curs hem vist que el problema d'universalitat de gramàtiques és indecidible, fins i tot restringit a l'alfabet $\{a, b\}$:

   $$U = \{G \in \mathtt{CFG}_{a,b} \mid \mathcal{L}(G) = \{a, b\}^*\}$$

   Considerem aquí el problema d'universalitat després de tallar el llenguatge representat per la gramàtica:

   $$UT = \{G \in \mathtt{CFG}_{a,b} \mid T(\mathcal{L}(G)) = \{a, b\}^*\}$$

   Demostreu raonadament que UT és indecidible amb una reducció des d'U.

Resposta:

(a) **Reduïm $\bar{K}$ a L per a demostrar que no és semi-decidible:**

$$x \mapsto \left\langle p = \left[\begin{array}{l}\textbf{Entrada } w\\ \quad \textbf{Executar } M_x(x)\\ \quad \textbf{Acceptar}\end{array}\right], q = \left[\begin{array}{l}\textbf{Entrada } w\\ \quad \textbf{Rebutjar}\end{array}\right]\right\rangle$$

És evident que la transformació és computable i total. Es preserva la resposta afirmativa perquè, si x pertany a $\bar{K}$, llavors $M_x(x)\uparrow$, i per tant $\mathcal{L}_p = \mathcal{L}_q = \emptyset$, de manera que $T(\mathcal{L}_q) = \emptyset = \mathcal{L}_p$, així que $\langle p, q\rangle \in L$. També es preserva la resposta negativa perquè, si x no pertany a $\bar{K}$, llavors $M_x(x)\downarrow$, i per tant $\mathcal{L}_p = \{a, b\}^*$ i $\mathcal{L}_q = \emptyset$, de manera que $T(\mathcal{L}_q) = \emptyset \neq \mathcal{L}_p$, així que $\langle p, q\rangle \notin L$.

(b) **Demostrarem $L = \mathbb{N}$, és a dir que tot p cumpleix la condició que defineix L, i això conclourà que L és decidible, perquè la següent màquina decideix $\mathbb{N}$:** `Entrada` p `Acceptar`. **Així doncs, sigui p un natural qualsevol. Sigui q el natural que codifica el programa següent:**

Entrada w
  Si $w = \lambda$ **llavors Rebutjar**
  Sigui v **el sufix de mida** $|w| - 1$ **de** w
  Si $M_p(v)$ **accepta llavors Acceptar**
  Rebutjar

Per a acabar, n'hi ha prou amb justificar $\mathcal{L}_p = T(\mathcal{L}_q)$. Però això és clar perquè que $M_p(v)$ accepti implica que tant $M_q(av)$ com $M_q(bv)$ accepten, i que qualsevol d'entre $M_q(av)$ o $M_q(bv)$ accepti, implica que $M_p(v)$ accepta.

(c) **Proposem com a reducció la transformació següent:**

$$G = \langle V, \{a, b\}, \delta, S\rangle \;\mapsto\; G' = \langle V \uplus \{S'\}, \{a, b\}, \delta \cup \{S' \to aS\}, S'\rangle$$

És evident que la transformació és computable i total. A més a més, $\mathcal{L}(G) = T(\mathcal{L}(G'))$. Per tant, $\mathcal{L}(G) = \{a, b\}^* \Leftrightarrow T(\mathcal{L}(G')) = \{a, b\}^*$, de manera que aquesta transformació preserva la resposta entre ambdós problemes.

(Examen de Juny-2012)

Aquest enunciat d'examen consta de dues parts. En una primera part hi ha una llista de llenguatges, on cada llenguatge ve marcat per una nota/puntuació. En una segona part hi ha els problemes de l'examen. En cada problema es demana escollir un llenguatge de la llista que compleixi una certa propietat, i demostrar que efectivament compleix aquesta propietat. La puntuació màxima de cada problema serà la del corresponent llenguatge escollit. Naturalment, si el llenguatge escollit no compleix la propietat demanada pel problema, la nota en aquell problema serà un 0. És perfectament possible que la puntuació total obtinguda en l'examen superi 10. En tal cas, la nota de l'examen serà 10. **Per a cada problema heu d'escollir un únic llenguatge: en cas contrari la nota d'aquell problema serà un 0. Fixeu-vos que els mateixos problemes ja indiquen un llenguatge vàlid, però de poca puntuació. Tingueu en compte que, a més puntuació, més dificultat.**

Llista de llenguatges:

1. (0.75 punts) $\{a^ib^j \mid i=j \ \wedge \ i=2j\}$ (opció vàlida problema 1)

2. (0.85 punts) $\{w \in \{a,b\}^* \mid \forall x,y : (w = xay \Rightarrow |y| \in \dot{2})\}$ (opció vàlida problema 2)

3. (1 punt) $\{a^ib^jc^kd^t \mid j = i+k+t\}$ (opció vàlida problema 3)

4. (0.85 punts) $\{p \mid \exists x : M_p(x) = 3\}$ (opció vàlida problema 4)

5. (0.85 punts) $\{p \mid \forall x : M_p(x) \uparrow\}$ (opció vàlida problema 5)

6. (0.85 punts) $\{\langle u,v,w,R\rangle \mid u \to_R^* v \ \wedge \ w \not\to_R^* v\}$ (opció vàlida problema 6)

7. (0.85 punts) $\{\langle G_1,G_2\rangle \in \mathtt{CFG}\times\mathtt{CFG} \mid \mathcal{L}(G_1)\cap\mathcal{L}(G_2) = \Sigma^*\}$ (opció vàlida problema 7)

8. (1.3 punts) $\{a^{n_0}ba^{n_1}b\dots ba^{n_k} \mid k \geqslant 1 \ \wedge \ \exists i,j : (0 \leqslant i < j \leqslant k \ \wedge \ n_i = n_j)\}$

9. (1.2 punts) $\{w \in \{a,b\}^* \mid \forall x,y : (w = xay \Rightarrow |ay|_{ab} > 0)\}$

10. (1.2 punts) $\{w \in \{a,b\}^* \mid \forall x,y : (w = xay \Rightarrow |y|_{ab} > 0)\}$
    (10bis)(1.2punts) $\{w \in \{a,b\}^* \mid |w|_a > 0 \wedge \forall x,y : (w = xay \Rightarrow |y|_{ab} > 0)\}$

11. (1.2 punts) $\{G \in \mathtt{CFG} \mid \mathcal{L}(G) = (ab+ba)^*\}$

12. (1.2 punts) $\{\langle p,q\rangle \mid \forall z : (\varphi_p(z)\downarrow \ \wedge \varphi_q(z)\downarrow \ \wedge \varphi_p(z) < \varphi_q(z))\}$

13. (1.3 punts) $\{\langle u,v,R\rangle \mid u \not\to_R^* v \ \wedge \ v \not\to_R^* u \ \wedge \ \exists w : (u \to_R^* w \ \wedge \ v \to_R^* w)\}$

14. (1.2 punts) $\{\langle p,q\rangle \mid \exists z : (\varphi_p(z)\downarrow \ \wedge \varphi_q(z)\downarrow \ \wedge \varphi_p(z) < \varphi_q(z))\}$

15. (2 punts) $\{p \mid \nexists q < p : \mathcal{L}_q = \mathcal{L}_p \cap \{1,2,3\}\}$

16. (2.25 punts) $\{p \mid \exists q < p : \mathcal{L}_q \cap \{1,2,3\} = \mathcal{L}_p\}$

17. (1.75 punts) $\{vu \mid \exists w \in \{a,b\}^* : (uv = ww^R)\}$

18. (2.25 punts) $\{R \mid \exists v : \forall u \in \Sigma^+ : u \to_R^* v\}$ (Σ és l'alfabet del sistema de reescriptura R)

19. (2 punts) $\{G \in \mathtt{CFG} \mid \overline{\mathcal{L}(G)}$ és finit$\}$ (el complementari s'interpreta respecte del conjunt de mots formats sobre l'alfabet de terminals de G)

20. (2.25 punts) $\{w \in \{0,1\}^* \mid \mathtt{valor}_2(ww) \in \dot{3}\}$

21. (2 punts) $f(K)$, on f és una funció computable, total i injectiva.

Llista de problemes:

1. Escolliu un llenguatge L de la llista que sigui finit (per exemple el llenguatge 1 compleix aquesta propietat). Demostreu que, efectivament, L és finit.

2. Escolliu un llenguatge L de la llista que sigui regular però no finit (per exemple el llenguatge 2 compleix aquesta propietat). No cal que demostreu que L no és finit. Dibuixeu el DFA mínim per a L. Justifiqueu que el DFA que heu dibuixat reconeix L de la manera que cregueu més convenient, ja sigui explicant el significat de cada estat, o per construcció: obtenint el DFA per operacions de tancament sobre llenguatges regulars simples, que provenent de la descomposició de la formalització de L. Argumenteu que el DFA que heu dibuixat és mínim, ja sigui amb l'algorisme de minimització, o argumentant que els estats són dos a dos distingibles, tot indicant simplement quins mots serien suficients per verificar totes les distingibilitats.

3. Escolliu un llenguatge L de la llista que sigui incontextual, però no regular (per exemple el llenguatge 3 compleix aquesta propietat). No cal que demostreu que L no és regular. Escriviu una CFG per a L. Justifiqueu que la CFG que heu escrit genera L de la manera que cregueu més convenient (indiqueu almenys quin llenguatge genera cada variable).

4. Escolliu un llenguatge L de la llista que sigui semi-decidible, però no decidible, i tal que es pugui mostrar una reducció "directa" des de K a L (per exemple el llenguatge 4 compleix aquesta propietat). Demostreu que L és semi-decidible, o bé amb un programa que semi-decideix L, o bé amb el teorema de projecció. Demostreu que L no és decidible amb la reducció esmentada.

5. Escolliu un llenguatge L de la llista que no sigui semi-decidible, i tal que pogueu mostrar una reducció "directa" des de $\bar{K}$ a L (per exemple el llenguatge 5 compleix aquesta propietat). Demostreu que L no és semi-decidible amb la reducció esmentada.

6. Escolliu un llenguatge L de la llista que no sigui decidible, i tal que pogueu mostrar una reducció "directa" des d'accessibilitat de mots ($\{\langle u, v, R\rangle \mid u \to_R^* v\}$) a L (per exemple el llenguatge 6 compleix aquesta propietat). Demostreu que L no és decidible amb la reducció esmentada Si ho considereu convenient, podeu assumir que l'alfabet de partida és $\{a, b\}$, i que la codificació de l'entrada exclou l'aparició de mots buits, fins i tot en els costats de les regles de reescriptura.

7. Escolliu un llenguatge L de la llista que no sigui decidible, i tal que pogueu mostrar una reducció "directa" des d'universalitat de gramàtiques ($\{G \in \texttt{CFG} \mid \mathcal{L}(G) = \Sigma^*\}$) a L (per exemple el llenguatge 7 compleix aquesta propietat). Demostreu que L no és decidible amb la reducció esmentada. Si ho considereu convenient, podeu assumir que l'alfabet de partida és $\{a, b\}$.

Nota: entenem que una reducció és directa si queda descrita de manera curta i sense fer servir composició de reduccions.

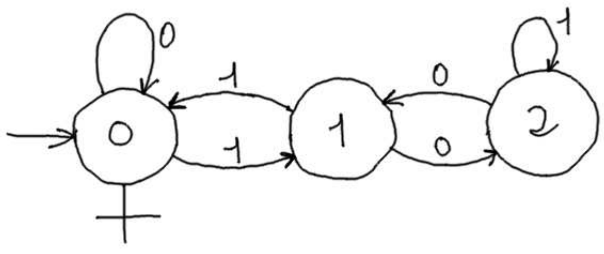
Resposta:

1. **Escollim el llenguatge L definit a 15 (també podiem haver escollit 1 i 10bis).**

   Per a cada subconjunt S de $\{1, 2, 3\}$, sigui p_S el nombre més petit tal que $\mathcal{L}_{p_S} = S$, és a dir, que p_S és el nombre més petit d'entre aquells que codifiquen programes tals que el seu llenguatge reconegut és S. Sigui P el màxim de tots aquests p_S. Amb aquestes definicions és clar que, per a tot p major que P, existeix q menor que p tal que $\mathcal{L}_q = \mathcal{L}_p \cap \{1, 2, 3\}$. En conclusió, cap p major que P pertany a L, i per tant L és finit.

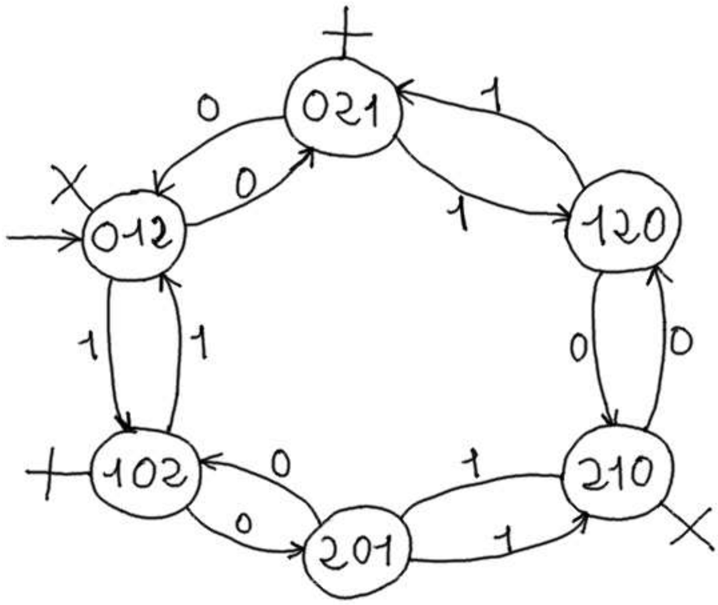
2. **Escollim el llenguatge L definit a 20 (també podiem haver escollit 2, 9 i 10).**

   Recordem el DFA mínim $A_{\dot{3}}$ que reconeix els múltiples de 3 ($\{w \mid \texttt{valor}_2(w) \in \dot{3}\}$):



   Per a reconéixer el llenguatge L construirem un DFA A que anirà fent tres simulacions d'execució del DFA $A_{\dot{3}}$: simularà l'execució d'$A_{\dot{3}}$ començant des de l'estat 0, simularà

l'execució d'$A_{\dot{3}}$ començant des de l'estat 1, i simularà l'execució d'$A_{\dot{3}}$ començant des de l'estat 2. Construïm A tenint, com a conjunt d'estats, tripletes $\langle e_0, e_1, e_2\rangle$ amb $e_0, e_1, e_2 \in \{0, 1, 2\}$. Definim l'estat inicial i les transicions de manera que e_0 és l'estat al qual s'hi arriba en $A_{\dot{3}}$ començant l'execució des de l'estat 0, e_1 és l'estat al qual s'hi arriba en $A_{\dot{3}}$ començant l'execució des de l'estat 1, i e_2 és l'estat al qual s'hi arriba en $A_{\dot{3}}$ començant l'execució des de l'estat 2. Així doncs, l'estat inicial és $\langle 0, 1, 2\rangle$, i les transicions es defineixen com $\delta_A(\langle e_0, e_1, e_2\rangle, c) = \langle \delta_{A_{\dot{3}}}(e_0, c), \delta_{A_{\dot{3}}}(e_1, c), \delta_{A_{\dot{3}}}(e_2, c)\rangle$ per a qualsevol símbol c. Per tal de reconéixer el llenguatge L, hem de posar com a acceptadors a aquells estats $\langle e_0, e_1, e_2\rangle$ tals que $e_{e_0} = 0$, ja que per a qualsevol mot w, i anomenant $\langle e_0, e_1, e_2\rangle = \delta_A(\langle 0, 1, 2\rangle, w)$, es compleix $\delta_{A_{\dot{3}}}(0, ww) = \delta_{A_{\dot{3}}}(\delta_{A_{\dot{3}}}(0, w), w) = \delta_{A_{\dot{3}}}(e_0, w) = e_{e_0}$. Aquest és el resultat:



Els mots $\lambda, 1, 0, 10$ són suficients per a comprovar que els estats són distingibles dos a dos.

3. **Escollim el llenguatge L definit a 17 (també podiem haver escollit 3 i 8).**

   Argumentem que L és $L' = \{uu^R vv^R \mid u, v \in \{a, b\}^*\}$. És clar que L' està inclòs a L, doncs "shiftant" qualsevol mot de la forma $uu^R vv^R$ podem obtenir $u^R vv^R u = u^R v(u^R v)^R$. Per a veure $L \subseteq L'$ considerem un mot w qualsevol de L. Per la definició de L, w és de la forma vu, on uv és de la forma ss^R. Considerarem dos casos: o bé (i) $|u| \geqslant |s|$ o bé (ii) $|u| < |s|$.

   En el cas (i), s és de la forma $s_1 s_2$, on $u = ss_2^R$ i $v = s_1^R$. Per tant, $w = vu = s_1^R ss_2^R = s_1^R s_1 s_2 s_2^R = s_1^R (s_1^R)^R s_2 s_2^R$. Això conclou que w pertany a L'.

   En el cas (ii), s és de la forma $s_1 s_2$, on $u = s_1$ i $v = s_2 s^R = s_2 s_2^R s_1^R$. Per tant, $w = vu = s_2 s_2^R s_1^R s_1 = s_2 s_2^R s_1^R (s_1^R)^R$. Això conclou que w pertany a L'.

   Sabent $L = \{uu^R vv^R \mid u, v \in \{a, b\}^*\}$, és fàcil obtenir una CFG G per a L:

   $S \to XX$

   $X \to aXa \mid bXb \mid \lambda$

   Notem que $\mathcal{L}(G, X) = \{ww^R \mid w \in \{a, b\}^*\}$.

4. **Escollim el llenguatge $f(K)$ definit a 21 (també podiem haver escollit 4 i 14).**

Justifiquem primer que la funció f és una reducció de K a $f(K)$. Per definició, és una transformació computable i total. Només falta veure que es preserva la resposta entre ambdós problemes, és a dir $x \in K \Leftrightarrow f(x) \in f(K)$. El fet $x \in K \Rightarrow f(x) \in f(K)$ és obvi per definició de conjunt imatge. El fet $x \notin K \Rightarrow f(x) \notin f(K)$ és degut a que f és injectiva: si suposem $x \notin K$ i $f(x) \in f(K)$, per definició de conjunt imatge, existeix $y \in K$ tal que $f(y) = f(x)$, i per la injectivitat se segueix $y = x$, cosa que contradiu $x \notin K$ i $y \in K$.

Donat que K no és decidible, i que f és una reducció de K a $f(K)$, concloem que $f(K)$ no és decidible.

Manca demostrar que $f(K)$ sí és semi-decidible. Donat que f és computable, considerem una màquina M_f que computa f. En conseqüència, la següent màquina semi-decideix $f(K)$:

```
Entrada y
  x:=0
  executar indefinidament:
    Si M_f(x) = y llavors:
      executar M_x(x)
      acceptar
    x++
```

Notem que $M_f(x) = y$ es comprova en temps finit perquè f és total. Donat un y qualsevol, si y és de $f(K)$, llavors existeix x de K tal que $f(x) = y$. Per a aquest x, $M_f(x)$ dona y, i $M_x(x)$ s'atura. Per tant, aquest y s'accepta, ja que, per la injectivitat de f, cap altre x' diferent de x complirà $f(x') = y$, i tard o d'hora el programa considerarà aquest x. En cas contrari, si l'y d'entrada no era de $f(K)$, llavors la màquina es penja, tant si no trobem cap x tal que $f(x) = y$, com si en trobem un, ja que en tal situació x no és de K, i per tant $M_x(x)$ es penja.

5. Escollim el llenguatge L definit a 16 (també podiem haver escollit 5 i 12).

   Sigui $p_\emptyset$ el menor nombre natural tal que la màquina que codifica reconeix el llenguatge buit, és a dir, $\mathcal{L}_{p_\emptyset} = \emptyset$. Per a cada nombre natural p, sigui I_p el nombre d'instruccions d'assignació que apareixen en el programa M_p. Sigui I el màxim de $I_0, I_1, I_2, \ldots, I_{p_\emptyset}$.

   Utilitzant les notacions anteriors, proposem la següent reducció des de $\bar{K}$:

$$x \mapsto p = \begin{cases} \textbf{Entrada y} \\ \quad y := 1 \\ \quad y := 2 \\ \quad y := 3 \\ \quad \vdots \\ \quad y := I \\ \quad y := I + 1 \\ \quad \textbf{Executar } M_x(x) \\ \quad \textbf{Acceptar} \end{cases}$$

   Notem que el corresponent p de cada x compleix que M_p té més assignacions que I. Això implica $p > p_\emptyset$

   És clar que la transformació és computable i total. Veiem que es preserva la resposta afirmativa d'esquerra a dreta suposant $x \in \bar{K}$. En tal cas, $L_p = \emptyset$. Per tant, $L_p = L_{p_\emptyset} \cap \{1, 2, 3\}$, i com que $p > p_\emptyset$, concloem $p \in L$.

Veiem que es preserva la resposta negativa d'esquerra a dreta suposant $x \notin \bar{K}$. En tal cas, $L_p = \mathbb{N}$. Per tant, cap $q < p$ compleix $L_p = L_q \cap \{1, 2, 3\}$. Això conclou $p \notin L$.

6. Escollim el llenguatge L definit a 18 (també podiem haver escollit 6 i 13).

   Proposem la següent reducció:

$$\langle u, v, R\rangle \;\mapsto\; R' = \begin{array}{l} R \cup \{c \to \$\#u\#\$,\ \#v\# \to @ \mid c \in (\Sigma \cup \{\#, \$\})\} \cup \\ \{c@ \to @, @c \to @ \mid c \in (\Sigma \cup \{\#, \$, @\})\} \end{array}$$

   on Σ és l'alfabet utilitzat en u, v, R, i $\#, \$, @$ són símbols nous.

   És clar que la transformació és computable i total. Veiem que es preserva la resposta afirmativa d'esquerra a dreta. Sigui $\langle u, v, R\rangle$ tal que $u \to_R^* v$. Per a demostrar que el corresponent R' pertany a L, veiem que @ és un mot accessible des de qualsevol altre mot no buit construït sobre $\Sigma' = (\Sigma \cup \{\#, \$, @\})$. Així doncs, sigui w un mot no buit qualsevol sobre Σ'. Si w conté algun @, és clar que $w \to_{R'}^* @$. En cas contrari, $w = cw'$, per a algun símbol c de $\Sigma' - \{@\}$. Per tant, $w = cw' \to_{R'} \$\#u\#\$w' \to_R^* \$\#v\#\$w' \to_{R'} \$@\$w' \to_{R'}^* @$. Això conclou la demostració.

   Veiem que es preserva la resposta afirmativa de dreta a esquerra. Sigui $\langle u, v, R\rangle$ tal que el corresponent R' pertany a L. Per tant, existeix un mot accessible des de qualsevol altre mot no buit. Però aquest ha de ser necessàriament @, doncs és l'únic mot no reescribible en cap altre. En particular, existeix una derivació $\# \to_{R'}^* @$. Queda per demostrar $u \to_R^* v$.

   Abans de continuar, desmostrarem la següent propietat per a qualsevol derivació $\# \to_{R'}^* w$ sense ocurrències d'@: per a qualsevol submot de w de la forma $\#s\#$ amb $s \in \Sigma^*$, es compleix $u \to_R^* s$. Ho fem per inducció, suposant que la propietat és certa sobre $\# \to_{R'}^* w$, i considerant un pas més de reescriptura $\# \to_{R'}^* w \to_{R'} w'$, on continua sense aparéixer @. Per aquest últim motiu, en l'últim pas de reescriptura només es pot haver utilitzat o bé (i) una regla de R, o bé (ii) una regla de la forma $c \to \$\#u\#\$$ on $c \in (\Sigma \cup \{\#, \$\})$. En el cas (i), és obvi que w' continua complint la propietat. En el cas (ii), w és de la forma w_1cw_2, i w' és de la forma $w_1\$\#u\#\$w_2$. Considerem un submot qualsevol de w' de la forma $\#s\#$ amb $s \in \Sigma^*$. Si $\#s\#$ és submot de w_1 o de w_2, llavors, com que w compleix la propietat, es compleix $u \to_R^* s$. Si $\#s\#$ no és submot de w_1 ni de w_2, llavors, necessàriament $s = u$. En aquest cas, $u \to_R^* s$ és obvi.

   Un cop vista la propietat anterior, tornem a la derivació $\# \to_{R'}^* @$. En aquesta derivació hi ha un pas on apareix @ per primera vegada. En aquell pas s'ha fet servir la regla $\#v\# \to @$. Òbviament, $\#v\#$ és submot del mot accedit abans d'aplicar la regla. Per la propietat anterior, es compleix $u \to_R^* v$. Això conclou la demostració.

7. Escollim el llenguatge L definit a 19 (també podiem haver escollit 7 i 11).

   Proposem la següent reducció des d'universalitat:

   $G = \langle V, \Sigma, \Delta, S\rangle \;\mapsto\; G' = \langle V \uplus \{S'\}, \Sigma \uplus \{\#\}, \Delta \cup \{S' \to S\#S'|S\}, S'\rangle$

   És clar que la transformació és computable i total. Veiem que preserva la resposta afirmativa d'esquerra a dreta suposant $\mathcal{L}(G) = \Sigma^*$. En tal cas, és clar que $\mathcal{L}(G') = (\Sigma \uplus \{\#\})^*$: això és perquè qualsevol mot w de $(\Sigma \uplus \{\#\})^*$ es pot descriure de la forma $w = w_1\#w_2\#\ldots\#w_n$ amb $w_1, w_2, \ldots, w_n \in \Sigma^*$, i tenim una derivació $S' \to_{G'} S\#S' \to_{G'}^* w_1\#S' \to_{G'} w_1\#S\#S' \to_{G'}^* w_1\#w_2\#S' \ldots \to_{G'}^* w$. Per tant, $\overline{\mathcal{L}(G')} = \emptyset$, que és un conjunt finit. Això conclou la demostració.

   Veiem que es preserva la resposta negativa d'esquerra a dreta suposant $\mathcal{L}(G) \neq \Sigma^*$. En tal cas, existeix un mot $w \notin \mathcal{L}(G)$. Llavors, qualsevol mot de la forma $w\#w\#\ldots\#w$ no és de $\mathcal{L}(G')$. Per tant, $\overline{\mathcal{L}(G')}$ és infinit. Això conclou la demostració.

(Examen de Juny-2012 (GRAU))

Per tal d'afavorir als estudiants honestos, en aquest examen, deixar en blanc una pregunta que no se sap respondre puntua més que contestar-la de qualsevol manera per "rascar". Cada pregunta o apartat va acompanyat de la seva valoració quan no es presenta o es deixa en blanc. Però la puntuació en blanc no pot aportar més d'una cinquena part de la nota que s'obtindria sumant punts de la forma tradicional deixant a 0 les respostes en blanc, és a dir:

$\texttt{Nota\_examen} = \texttt{Nota\_tradicional} + \texttt{minim}(\frac{\texttt{Nota\_tradicional}}{5}, \texttt{Nota\_en\_blanc})$

CONCEPTE PRELIMINAR:

Donat un llenguatge L, *shiftar* L dóna lloc a un nou llenguatge, que denotem $S(L)$, i que conté als mots que s'obtenen agafant cada mot de L i rotant-lo de totes les maneres possibles.

$$S(L) = \{vu \mid uv \in L\}$$

Per exemple, $S(\{\lambda, a, bb, baa, abab\}) = \{\lambda, a, bb, baa, aba, aab, abab, baba\}$, $S(a^*) = a^*$ i $S(\emptyset) = \emptyset$.

PREGUNTES:

1. (a) (0.75 punts, en blanc 0.1, dificultat estimada: baixa, Temps de resolució estimat: 20 minuts) Justifiqueu si les següents igualtats referides a un llenguatge L qualsevol són certes (amb una demostració) o falses (amb un contraexemple).
      i. $S(\overline{L}) = \overline{S(L)}$.
      ii. $S(L^R) = (S(L))^R$.

   (b) (0.5 punts, en blanc 0.1, dificultat estimada: baixa, Temps de resolució estimat: 10 minuts) Sigui L el conjunt de mots sobre $\{a, b\}$ que comencen per a, formalment: $L = \{w \in \{a, b\}^* \mid \exists w' : w = aw'\}$. Formalitzeu el llenguatge $S(L)$ amb notació clàssica de conjunts, sense més explicació. A més a més, dibuixeu el DFA mínim que reconeix $S(L)$. No cal cap explicació sobre l'autòmat ni justificar que és mínim.

   (c) (0.5 punts, en blanc 0.1, dificultat estimada: baixa, Temps de resolució estimat: 10 minuts) Per al DFA del llenguatge $S(L)$ de l'apartat anterior, mostreu com s'hi aplica el lema d'Arden per a obtenir una expressió regular que representi $S(L)$.

   (d) (0.5 punts, en blanc 0.1, dificultat estimada: baixa, Temps de resolució estimat: 10 minuts) Sigui $L = \{a^n b^n \mid n \geqslant 0\}$. Formalitzeu el llenguatge $S(L)$ amb notació clàssica de conjunts, sense més explicació (si voleu, podeu representar $S(L)$ com a unió de dos o més conjunts). A més a més, escriviu una CFG no ambigua que genera $S(L)$. No cal cap explicació sobre la gramàtica.

   (e) (0.5 punts, en blanc 0.1, dificultat estimada: baixa, Temps de resolució estimat: 15 minuts) Dibuixeu un autòmat amb pila determinista que reconegui el llenguatge $S(L)$ de l'apartat anterior. No cal cap explicació.

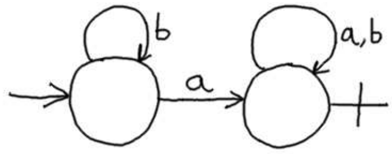
Resposta:

(a)

i. La igualtat $S(\overline{L}) = \overline{S(L)}$ és falsa. Per exemple, el mot ab pertany a $S(\overline{\{ab\}})$ però no pertany a $\overline{S(\{ab\})}$.

ii. La igualtat $S(L^R) = (S(L))^R$ és certa: $S(L^R) = \{vu \mid uv \in L^R\} = \{vu \mid (uv)^R \in L\} = \{vu \mid v^R u^R \in L\} = \{v^R u^R \mid vu \in L\} = \{(uv)^R \mid vu \in L\} = \{uv \mid vu \in L\}^R = (S(L))^R$.

(b) $S(L) = \{w \in \{a,b\}^* \mid \exists w_1, w_2 : w = w_1 a w_2\}$**.**



(c)

$$
\begin{aligned}
X_0 &= bX_0 + aX_1 \\
X_1 &= aX_1 + bX_1 + \Lambda
\end{aligned}
$$

$$
\begin{aligned}
X_1 &= (a+b)X_1 + \Lambda \\
X_1 &= (a+b)^* \\
X_0 &= bX_0 + a(a+b)^* \\
X_0 &= b^*a(a+b)^*
\end{aligned}
$$

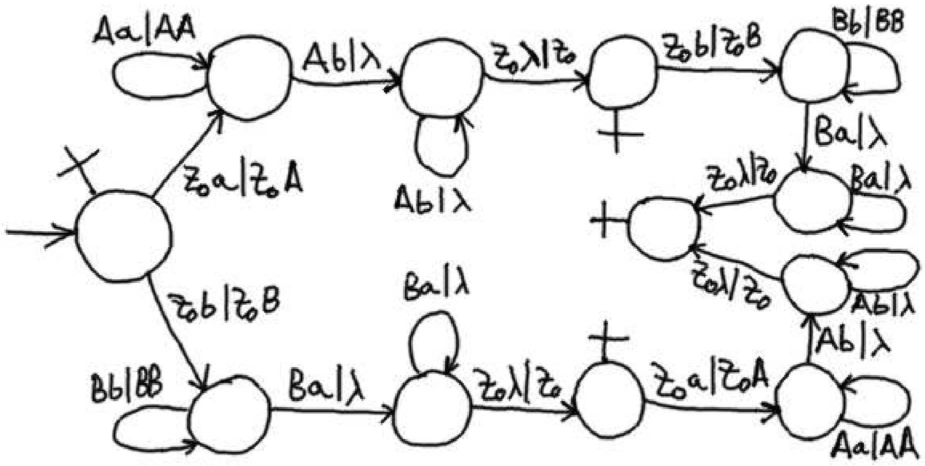
Per tant, $b^*a(a+b)^*$ **representa** $S(L)$**.**

(d)

$$S(L) = \{b^n a^m b^k \mid m = n+k\} \cup \{a^n b^m a^k \mid m = n+k\}$$

$$
\begin{aligned}
S &\rightarrow aXbY \mid bYaX \mid \lambda \\
X &\rightarrow aXb \mid \lambda \\
Y &\rightarrow bYa \mid \lambda
\end{aligned}
$$

(e)



2. (a) (1 punt, en blanc 0.15, dificultat estimada: mitja, Temps de resolució estimat: 15 minuts) Demostreu que la implicació (L decidible $\Rightarrow$ $S(L)$ decidible) és certa per a qualsevol llenguatge L.

(b) (1 punt, en blanc 0.15, dificultat estimada: mitja, Temps de resolució estimat: 15 minuts) Demostreu que la implicació (L semi-decidible $\Rightarrow S(L)$ semi-decidible) és certa per a qualsevol llenguatge L.

(c) (1.5 punts, en blanc 0.15, dificultat estimada: alta, Temps de resolució estimat: 25 minuts) Demostreu que la implicació (L regular $\Rightarrow S(L)$ regular) és certa per a qualsevol llenguatge L.

Pista: Donat un DFA A que reconeix L, i dos estats q_1, q_2 d'A, definim L_{A,q_1,q_2} com el conjunt de mots que ens porten des de l'estat q_1 fins l'estat q_2 executant A. És evident que aquests llenguatges L_{A,q_1,q_2} són regulars. Us recomanem que intenteu descriure $S(L)$ en termes d'aquests llenguatges i operacions tancades sobre llenguatges regulars.

Resposta:

(a) Sigui L un llenguatge decidible. Llavors existeix una màquina M_L que decideix L. En conseqüència, la següent màquina decideix $S(L)$:

Entrada w
  Per cada partició $w = uv$ **fer:**
    Si $M_L(vu)$ **accepta llavors Acceptar**
  Rebutjar

Notem que la màquina para amb tota entrada perquè el bucle considera un nombre finit de casos i M_L para amb tota entrada. A més a més, és clar que reconeix $S(L) = \{vu \mid uv \in L\}$.

(b) Sigui L un llenguatge semi-decidible. Llavors existeix una màquina M_L que semi-decideix L. En conseqüència, la següent màquina semi-decideix $S(L)$:

Entrada w
  t**:=1**
  Executar indefinidament:
    Per cada partició $w = uv$ **fer:**
      Si $M_L(vu)$ **accepta en** t **passos llavors Acceptar**
    t**++**

Si una entrada w és de $S(L)$, llavors existeixen u, v tals que $w = uv$ i $vu \in L$. Per tant, $M_L(vu)$ accepta en un cert nombre de passos T. Quan la variable t de la màquina anterior arribi a valer T, es considerarà la partició $w = uv$ i la màquina acceptarà.

En el cas d'una entrada w que no sigui de $S(L)$, la màquina anterior es penja, doncs no es trobarà mai un cas de u, v on $w = uv$ i tal que $M_L(vu)$ accepta. Això conclou que la màquina anterior semi-decideix $S(L)$.

(c) Sigui L un llenguatge regular qualsevol. Sigui $A = \langle Q, \Sigma, \delta, q_0, F\rangle$ un DFA que reconeix L. Observem que un mot de $S(L)$ és un mot de la forma vu, per a uns certs u, v tals que uv pertany a L. Per tant, el mot u ens porta des de l'estat inicial q_0 fins a un cert estat intermig q, i el mot v ens porta des de q fins a un cert estat acceptador q_f. Així doncs, el llenguatge $S(L)$ es pot descriure com $S(L) = \bigcup_{q\in Q, q_f \in F}(L_{q,q_f,A} \cdot L_{q_0,q,A})$. Donat que aquesta expressió consta d'unions i concatenacions sobre llenguatges regulars, concloem que $S(L)$ és regular.

3. Per al primer subapartat d'aquest problema, recordeu que, tot i que se suposa que els programes reben naturals com a entrada, quan calgui es pot assumir que aquestes entrades codifiquen mots sobre $\{a, b\}^*$ de forma biunívoca.

(a) (1.25 punts, en blanc 0.15, dificultat estimada: mitja, Temps de resolució estimat: 20 minuts) Classifiqueu raonadament com a decidible, semi-decidible però no decidible, o no semi-decidible, el següent conjunt:

$$L = \{p \mid \mathcal{L}_p = S(\mathcal{L}_p)\}$$

(b) (1.25 punts, en blanc 0.15, dificultat estimada: mitja, Temps de resolució estimat: 15 minuts) Considerem els següents problemes/conjunts, on s'assumeix que ni u ni v ni cap costat de regla de R és el mot buit (de fet s'assumeix que la pròpia codificació de l'entrada exclou aquests casos):

`Accessibilitat` (decidir si un mot u pot accedir a un mot v aplicant regles de reescriptura d'un conjunt R):

$$A = \{\langle u, v, R\rangle \mid u \to_R^* v\}$$

`Accessibilitat-shiftada` (decidir si un mot u pot accedir a alguna de les possibles maneres de shiftar un mot v, a base d'aplicar regles de reescriptura d'un conjunt R):

$$A_{\texttt{shift}} = \{\langle u, v, R\rangle \mid \exists v_1, v_2 : (v = v_1 v_2 \wedge u \to_R^* v_2 v_1)\}$$

Demostreu que $A_{\texttt{shift}}$ is indecidible amb una reducció des d'A.

(c) (1.25 punts, en blanc 0.15, dificultat estimada: mitja, Temps de resolució estimat: 25 minuts) Durant el curs hem vist que el problema d'intersecció no buida de llenguatges generats per gramàtiques és indecidible:

$$I = \{\langle G_1, G_2\rangle \in \texttt{CFG} \mid \mathcal{L}(G_1) \cap \mathcal{L}(G_2) \neq \emptyset\}$$

Considerem aquí el problema d'intersecció no buida després de shiftar els llenguatges representats per les gramàtiques:

$$IS = \{\langle G_1, G_2\rangle \in \texttt{CFG} \mid S(\mathcal{L}(G_1)) \cap S(\mathcal{L}(G_2)) \neq \emptyset\}$$

Demostreu raonadament que IS és indecidible amb una reducció des d'I.

Resposta:

(a) Reduïm $\bar{K}$ a L per a demostrar que L no és semi-decidible:

$$x \mapsto p = \left[\begin{array}{l} \textbf{Entrada } w \\ \quad \textbf{Executar } M_x(x) \\ \quad \textbf{Si } w = ab \textbf{ llavors Acceptar} \\ \quad \textbf{Rebutjar} \end{array}\right]$$

És evident que la transformació és computable i total. Es preserva la resposta afirmativa perquè, si x pertany a $\bar{K}$, llavors $M_x(x)\uparrow$, i per tant $\mathcal{L}_p = \emptyset$, de manera que $S(\mathcal{L}_p) = \emptyset = \mathcal{L}_p$, així que $p \in L$. També es preserva la resposta negativa perquè, si x no pertany a $\bar{K}$, llavors $M_x(x)\downarrow$, i per tant $\mathcal{L}_p = \{ab\}$, de manera que $S(\mathcal{L}_p) = \{ab, ba\} \neq \mathcal{L}_p$, així que $p \notin L$.

(b) Proposem com a reducció la transformació següent:

$$\langle u, v, R\rangle \;\mapsto\; \langle u, \#, R' = R \cup \{v \to \#\}\rangle$$

on # és un símbol nou.

És clar que la transformació és computable i total. Veiem que preserva la resposta afirmativa suposant $u \to_R^* v$. Llavors, tenim $u \to_R^* v \to_{v\to\#} \#$, i per tant $u \to_{R'}^* \#$. Així doncs, u accedeix amb R' a una de les possibles maneres de shiftar $\#$ (de fet només n'hi ha una). Això conclou la demostració de la resposta afirmativa.

Ara, veiem que es preserva la resposta negativa suposant que u accedeix amb R' a alguna de les possibles maneres de shiftar $\#$. Com que $\#$ és només un símbol, això implica $u \to_{R'}^* \#$. L'última regla aplicada ha de ser, necessàriament, $v \to \#$. Per tant, tenim $u \to_{R'}^* v$. En aquesta derivació, la regla $v \to \#$ no es pot haver utilitzat cap cop, perquè cap regla pot esborrar ocurrències del símbol $\#$, i aquest símbol no apareix cap vegada a v. Així doncs, totes les regles utilitzades a $u \to_{R'}^* v$ són de R. Això conclou $u \to_R^* v$.

(c) **Proposem com a reducció la transformació següent:**

$$\left\langle \begin{array}{c} G_1 = \langle V_1, \Sigma, \delta_1, S_1 \rangle, \\ G_2 = \langle V_2, \Sigma, \delta_2, S_2 \rangle \end{array} \right\rangle \mapsto \left\langle \begin{array}{c} G_1' = \langle V_1 \uplus \{S_1'\}, \Sigma \uplus \{\#\}, \delta_1 \cup \{S_1' \to \#S_1\}, S_1' \rangle, \\ G_2' = \langle V_2 \uplus \{S_2'\}, \Sigma \uplus \{\#\}, \delta_2 \cup \{S_2' \to \#S_2\}, S_2' \rangle \end{array} \right\rangle$$

És evident que la transformació és computable i total. A més a més, notem que $\mathcal{L}(G_1') = \#\mathcal{L}(G_1)$ i $\mathcal{L}(G_2') = \#\mathcal{L}(G_2)$. Queda veure que es preserva la resposta entre ambdós problemes.

Comencem demostrant la preservació de resposta afirmativa d'esquerra a dreta suposant $\mathcal{L}(G_1) \cap \mathcal{L}(G_2) \neq \emptyset$. Sigui w un mot de $\mathcal{L}(G_1) \cap \mathcal{L}(G_2)$. Llavors, $\#w$ és un mot de $\#\mathcal{L}(G_1) \cap \#\mathcal{L}(G_2)$, i per tant també de $S(\#\mathcal{L}(G_1)) \cap S(\#\mathcal{L}(G_2))$. En conseqüència, $S(\#\mathcal{L}(G_1)) \cap S(\#\mathcal{L}(G_2)) \neq \emptyset$, de manera que $S(\mathcal{L}(G_1')) \cap S(\mathcal{L}(G_2')) \neq \emptyset$,

Ara demostrem la preservació de resposta afirmativa de dreta a esquerra suposant $S(\mathcal{L}(G_1')) \cap S(\mathcal{L}(G_2')) \neq \emptyset$. Notem que això és equivalent a dir $S(\#\mathcal{L}(G_1)) \cap S(\#\mathcal{L}(G_2)) \neq \emptyset$. Sigui w un mot de $S(\#\mathcal{L}(G_1)) \cap S(\#\mathcal{L}(G_2))$. Com que $\#$ és un símbol nou, apareix un sol cop a w, i llavors, w és de la forma $w_1\#w_2$, per a certs w_1, w_2 que no contenen $\#$. A més a més, per la definició de shiftar un llenguatge, $\#w_2w_1$ també pertany a $S(\#\mathcal{L}(G_1))$ i a $S(\#\mathcal{L}(G_2))$. Però de fet, això implica que $\#w_2w_1$ pertany a $\#\mathcal{L}(G_1)$ i a $\#\mathcal{L}(G_2)$. En conseqüència, w_2w_1 pertany a $\mathcal{L}(G_1)$ i a $\mathcal{L}(G_2)$, de manera que $\mathcal{L}(G_1) \cap \mathcal{L}(G_2) \neq \emptyset$.

(Examen de Gener-2013)

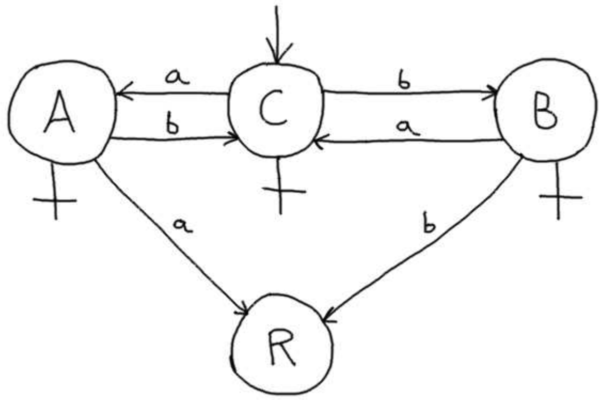
Per tal d'afavorir als estudiants honestos, en aquest examen, deixar en blanc una pregunta que no se sap respondre puntua més que contestar-la de qualsevol manera per "rascar". Cada pregunta o apartat va acompanyat de la seva valoració quan no es presenta o es deixa en blanc. Però la puntuació en blanc no pot aportar més d'una cinquena part de la nota que s'obtindria sumant punts de la forma tradicional deixant a 0 les respostes en blanc, és a dir:

$$\texttt{Nota\_examen} = \texttt{Nota\_tradicional} + \texttt{minim}(\tfrac{\texttt{Nota\_tradicional}}{5}, \texttt{Nota\_en\_blanc})$$

1. (2.5 punts en total) Obteniu raonadament els DFAs mínims que reconeixen els següents llenguatges:
   (a) (1.25 punts, en blanc: 0.15) $L_1 = \{w \in \{a,b\}^* \mid \forall u, v : (w = uv \Rightarrow -1 \leqslant |u|_a - |u|_b \leqslant 1)\}$
   (b) (1.25 punts, en blanc: 0.15) $L_2 = \{w \in \{a,b\}^* \mid \forall u, v : (w = uv \Rightarrow -1 \leqslant |v|_a - |v|_b \leqslant 1)\}$

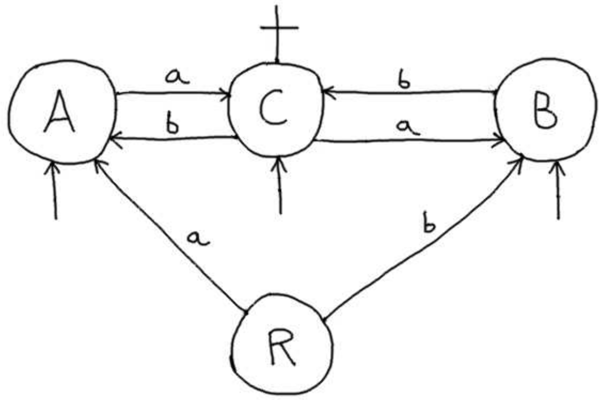
Resposta:

(a) **Ens cal un estat A al qual s'hi arriba amb mots de $L_1 \cap \{w \in \{a,b\}^* \mid |w|_a = 1 + |w|_b\}$, un estat B al qual s'hi arriba amb mots de $L_1 \cap \{w \in \{a,b\}^* \mid |w|_b = 1 + |w|_a\}$, un estat C al qual s'hi arriba amb mots de $L_1 \cap \{w \in \{a,b\}^* \mid |w|_a = |w|_b\}$, i un estat pou R al qual s'hi arriba amb mots de $\overline{L_1}$. D'aquesta manera obtenim el següent DFA A_1:**

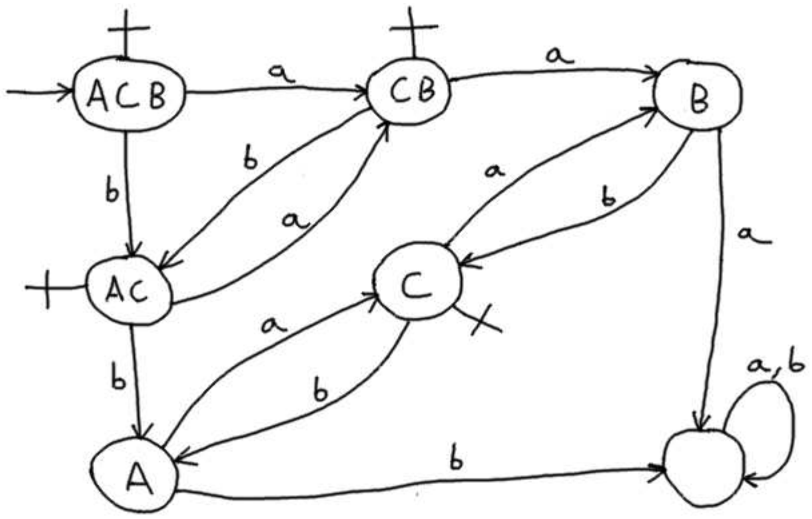


Els estats de A_1 són dos a dos distingibles perquè els mots a, b ens permeten distingir-los, i per tant A_1 és mínim.

(b) **Donat que L_2 és L_1^R, podem aplicar la operació de revessar autòmats sobre A_1 obtenint així un NFA A_2' per a L_2. Aquesta operació consisteix en: canviar la direcció de les fletxes, convertir en inicials els estats que eren acceptadors, i convertir en acceptadors els estats que eren inicials.**



Si determinitzem A_2' obtenim el següent DFA A_2:



El DFA A_2 és mínim perquè els mots a, b permeten veure que els estats són dos a dos distingibles.

2. (2.5 punts en total) Obteniu raonadament CFGs no ambigües (no cal que demostreu la no ambigüitat) que generin els següents llenguatges:

   (a) (1.25 punts, en blanc: 0.15) $L_3 = \{w \in \{a,b\}^* \mid |w| \in \dot{2} \land w = w^R \land \forall u, v : (w = uv \Rightarrow -1 \leqslant |u|_a - |u|_b \leqslant 1)\}$

   (b) (1.25 punts, en blanc: 0.15) $L_4 = \{w \in \{a,b\}^* \mid w = w^R \land \forall u, v : (w = uv \Rightarrow -1 \leqslant |u|_a - |u|_b \leqslant 1)\}$

Resposta:

(a) Notem que L_3 és $(ab+ba)^* \cap \{w \in \{a,b\}^* \mid w = w^R\}$. Una CFG no ambigua per a aquesta descripció s'obté fàcilment:

$$S \to abSba \mid baSab \mid \lambda$$

(b) Recordem que una CFG G per a $\{w \in \{a,b\}^* \mid w = w^R\}$ és $S \to aSa \mid bSb \mid a \mid b \mid \lambda$. Per a obtenir una CFG per a L_4, el que podem fer és modificar G per a que alhora vagi simulant l'execució de A_1 (el primer DFA del problema anterior) amb els símbols que va deixant a l'esquerra de S, i també vagi simulant A_1 però anant amb les fletxes cap enrera amb els símbols que va deixant a la dreta de S. D'aquesta manera, podrà verificar també la condició $\forall u, v : (w = uv \Rightarrow -1 \leqslant |u|_a - |u|_b \leqslant 1)$ sobre el mot generat, que és precisament la que comprova el DFA A_1. Així doncs, tindrem variables de la forma $S_{q_1 q_2}$, on q_1, q_2 són estats de A_1, i aquestes variables generaran mots w tals que $w = w^R$ i, a més a més, A_1 ens porta de q_1 a q_2 amb el mot w. Amb aquestes condicions, el nou símbol inicial S' haurà de tenir, com a parts dretes, les variables

$S_{q_1 q_2}$ tals que q_1 és l'estat inicial C de A_1, i q_2 és qualsevol dels estats acceptadors de A_1. D'aquesta manera obtenim la següent CFG:

$$
\begin{array}{l}
S' \to S_{CA} \mid S_{CC} \mid S_{CB} \\
S_{CA} \to aS_{AC}a \mid a \\
S_{CB} \to bS_{BC}b \mid b \\
S_{CC} \to aS_{AB}a \mid bS_{BA}b \mid \lambda \\
S_{AC} \to bS_{CA}b \mid b \\
S_{BC} \to aS_{CB}a \mid a \\
S_{AB} \to bS_{CC}b \\
S_{BA} \to aS_{CC}a
\end{array}
$$

Essencialment, aquest és un cas particular de la construcció per a obtenir una CFG que genera la intersecció d'un llenguatge incontextual i un llenguatge regular a partir d'una CFG i un DFA inicials donats. Donat que aquesta construcció preserva la no ambigüitat, i que la CFG $S \to aSa \mid bSb \mid a \mid b \mid \lambda$ és no ambigua, la CFG resultant és no ambigua.

3. (2.5 punts en total) Classifiqueu raonadament com a decidible, semi-decidible però no decidible, o no semi-decidible, els següents conjunts:

   (a) (1.25 punts, en blanc: 0.15) $C_1 = \{\langle p, q\rangle \mid \phi_p(q)\downarrow \,\wedge\, \phi_q(p)\downarrow \,\wedge\, \phi_p(q) = \phi_q(p)\}$.

   (b) (1.25 punts, en blanc: 0.15) $C_2 = \{\langle p, q\rangle \mid (\phi_p(q)\downarrow \,\wedge\, \phi_q(p)\downarrow) \Rightarrow \phi_p(q) \neq \phi_q(p)\}$.

Resposta:

(a) Demostrem que C_1 no és decidible amb la següent reducció des de K:

$$x \mapsto \left\langle p = \left[\begin{array}{l}\textbf{Entrada } y \\ \quad \textbf{Executar } M_x(x) \\ \quad \textbf{Sortida 1}\end{array}\right], q = \left[\begin{array}{l}\textbf{Entrada } y \\ \quad \textbf{Sortida 1}\end{array}\right]\right\rangle$$

És clar que la transformació és computable i total.
Donat $x \in K$, tenim que $M_x(x)\downarrow$, i llavors els corresponents p, q cumpleixen $\phi_p(q) = \phi_q(p) = 1$, de manera que $\langle p, q\rangle \in C_1$.
Donat $x \notin K$, tenim que $M_x(x)\uparrow$, i llavors els corresponents p, q cumpleixen $\phi_p(q)\uparrow$, de manera que $\langle p, q\rangle \notin C_1$.

En segon lloc veiem que C_1 és semi-decidible perquè el següent programa semi-decideix C_1:

```
Entrada ⟨p,q⟩
   t:=1
   executar indefinidament:
     Si M_p(q) i M_p(p) paren en t passos i M_p(q) = M_q(p) llavors:
       acceptar
     t++
```

(b) Notem que $C_2 = \overline{C_1}$. Donat que C_1 és semi-decidible però no decidible, podem concloure directament que C_2 no és ni semi-decidible, doncs sabem que quan un llenguatge i el seu complementari són alhora semi-decidibles, resulta que llavors han de ser també decidibles.

4. (2.5 punts en total) Considereu els següents conjunts, on s'assumeix implícitament que el rang de u, v, w és els mots no buits, i el de R és el dels sistemes de reescriptura amb regles $l \to r$ on ni l ni r són el mot buit. De fet, s'assumeix que la pròpia codificació de les tuples d'entrada exclou els casos amb mots buits.

$$A = \{\langle u, v, R\rangle \mid u \to_R^* v\}$$

$$A_1 = \{\langle u, v, R\rangle \mid uu \to_R^* vv\}$$

$$A_2 = \{\langle u, v, w, R\rangle \mid uu \to_R^* ww \wedge v \to_R^* ww\}$$

Durant el curs hem vist que A és indecidible.

(a) (1.25 punts, en blanc: 0.15) Demostreu que A_1 és indecidible amb una reducció des d'A (és a dir, $A \leqslant A_1$).

(b) (1.25 punts, en blanc: 0.15) Demostreu que A_2 és indecidible amb una reducció des d'A_1 (és a dir, $A_1 \leqslant A_2$).

Resposta:

(a) Proposem com a reducció la transformació següent:

$$\langle u, v, R\rangle \;\mapsto\; \langle \#u, \#v, R\rangle$$

on # és un símbol nou.

És clar que la transformació és computable i total. Veiem que preserva la resposta afirmativa suposant $\langle u, v, R\rangle \in A$. Llavors es cumpleix $u \to_R^* v$, i per tant, tenim $\#u\#u \to_R^* \#v\#v$, de manera que $\langle \#u, \#v, R\rangle \in A_1$.

Ara, veiem que es preserva la resposta negativa suposant $\langle \#u, \#v, R\rangle \in A_1$. Llavors, es cumpleix $\#u\#u \to_R^* \#v\#v$, i donat que # és un símbol nou, i que per tant no apareix a R, això implica $u \to_R^* v$. Per tant, concloem $\langle u, v, R\rangle \in A$.

(b) Proposem com a reducció la transformació següent:

$$\langle u, v, R\rangle \;\mapsto\; \langle u, uu, v, R\rangle$$

on # és un símbol nou.

És clar que la transformació és computable i total. A més a més, és clar que es preserva la resposta, ja que $uu \to_R^* vv \;\Leftrightarrow\; (uu \to_R^* vv \;\wedge\; uu \to_R^* vv)$.

(Examen de Gener-2013 (GRAU))

Per tal d'afavorir als estudiants honestos, en aquest examen, deixar en blanc una pregunta que no se sap respondre puntua més que contestar-la de qualsevol manera per "rascar". Cada pregunta o apartat va acompanyat de la seva valoració quan no es presenta o es deixa en blanc. Però la puntuació en blanc no pot aportar més d'una cinquena part de la nota que s'obtindria sumant punts de la forma tradicional deixant a 0 les respostes en blanc, és a dir:

$\texttt{Nota\_examen} = \texttt{Nota\_tradicional} + \texttt{minim}(\frac{\texttt{Nota\_tradicional}}{5}, \texttt{Nota\_en\_blanc})$

CONCEPTE PRELIMINAR:

Donat un mot w de $\{a,b,c\}^*$, *ordenar* w dona lloc a un nou mot, que denotem $O(w)$, i que s'obté ordenant alfabèticament els símbols de w, és a dir, posant primer totes les a's de w, després totes les b's de w, i finalment totes les c's de w. Més formalment, $O(w) = a^{|w|_a}b^{|w|_b}c^{|w|_c}$. Per exemple, $O(abacba) = aaabbc$, $O(bcba) = abbc$, $O(aabbbcc) = aabbbcc$, $O(ba) = ab$, $O(a) = a$ i $O(\lambda) = \lambda$.

Donat un llenguatge L sobre l'alfabet $\{a,b,c\}$, *ordenar* L dona lloc a un nou llenguatge, que denotem $O(L)$, i que s'obté ordenant tots els mots de L. Més formalment, $O(L) = \{O(w) \mid w \in L\}$. Per exemple, $O(\{abacba, bcba, aabbbcc, ba, a, \lambda\}) = \{aaabbc, abbc, aabbbcc, ab, a, \lambda\}$, $O(\{aa, ab, ba, bb\}) = \{aa, ab, bb\}$, $O(\{abb, bab, bba\}) = \{abb\}$, $O(\emptyset) = \emptyset$, $O(\{\lambda\}) = \{\lambda\}$ i $O(\{a,b,c\}^*) = \{a^n b^m c^k \mid n,m,k \geqslant 0\}$. Fixeu-vos en que si un llenguatge L és subconjunt de $\{a\}^*$ o de $\{a,b\}^*$, $O(L)$ també queda correctament definit.

PREGUNTES:

1. (3 punts en total)

   (a) (1 punt, en blanc 0.1, dificultat estimada: baixa, Temps de resolució estimat: 15 minuts) Justifiqueu si les següents igualtats referides a llenguatges L, L_1, L_2 qualssevol i a un morfisme $\sigma : \{a,b,c\}^* \to \{a\}^*$ qualsevol són certes (amb una demostració) o falses (amb un contraexemple). (Fixeu-vos que els mots de la imatge de σ contenen només a's.)

      i. $O(L_1 \cap L_2) = O(L_1) \cap O(L_2)$.

      ii. $O(\sigma(L)) = \sigma(O(L))$.

   (b) (1 punt, en blanc 0.1, dificultat estimada: mitjana/baixa, Temps de resolució estimat: 20 minuts) Sigui L el conjunt de mots capicua sobre $\{a,b\}$, formalment: $L = \{w \in \{a,b\}^* \mid w = w^R\}$. Formalitzeu el llenguatge $O(L)$ de manera senzilla amb notació clàssica de conjunts, sense utilitzar les operacions revessar i O. A més a més, dibuixeu el DFA mínim que reconeix $O(L)$. Expliqueu breument el que feu. No cal justificar que el DFA és mínim.

   (c) (1 punt, en blanc 0.1, dificultat estimada: mitjana/baixa, Temps de resolució estimat: 20 minuts) Sigui L el llenguatge dels mots w sobre $\{a,b,c\}$ tals que en posicions senars de w només hi ha a's (la primera posició del mot s'indexa amb 1). Formalment, $L = \{w \in \{a,b,c\}^* \mid \forall x,y : (w = xy \wedge |x| \in \dot{2} \wedge |y| > 0 \Rightarrow \exists y' : y = ay')\}$. Formalitzeu el llenguatge $O(L)$ amb notació clàssica de conjunts, sense més explicació. A més a més, escriviu una CFG no ambigua que genera $O(L)$. Expliqueu breument el que feu.

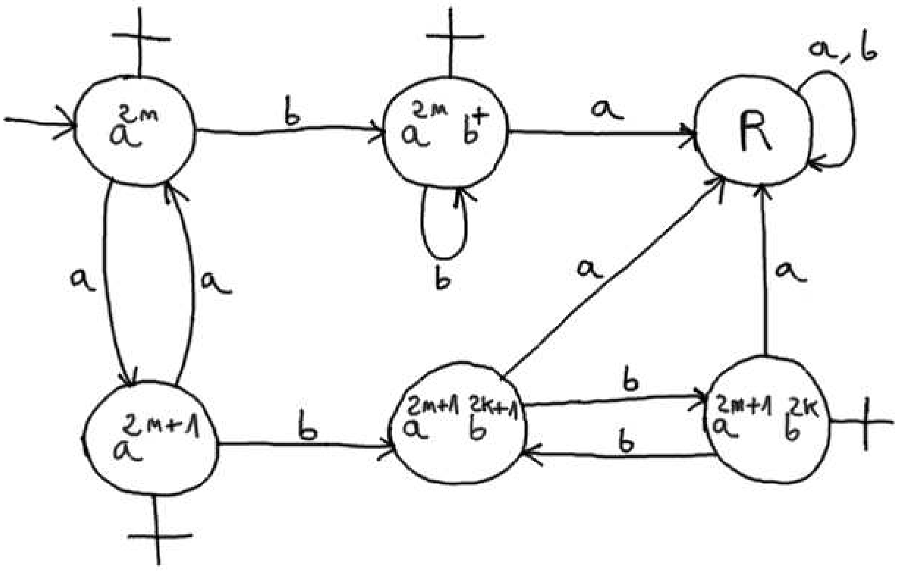
Resposta:

(a)

i. La igualtat $O(L_1 \cap L_2) = O(L_1) \cap O(L_2)$ és falsa. Per exemple, definint $L_1 = \{ab\}$, $L_2 = \{ba\}$ tenim que $O(L_1 \cap L_2) = O(\emptyset) = \emptyset$ però $O(L_1) \cap O(L_2) = \{ab\}$.

ii. **La igualtat $O(\sigma(L)) = \sigma(O(L))$ és certa per a un morfisme $\sigma : \{a,b,c\}^* \to \{a\}^*$. Per a veure-ho, ens convencem primer que $O(\sigma(w)) = \sigma(O(w))$ per a qualsevol mot $w \in \{a,b,c\}^*$. Donat que tant $O(\sigma(w))$ com $\sigma(O(w))$ són de a^*, per a justificar $O(\sigma(w)) = \sigma(O(w))$ n'hi ha prou amb veure $|O(\sigma(w))| = |\sigma(O(w))|$, cosa que obtenim amb la seqüència d'igualtats següent: $|O(\sigma(w))| = |\sigma(w)| = |\sigma(a)| \cdot |w|_a + |\sigma(b)| \cdot |w|_b + |\sigma(c)| \cdot |w|_c = |\sigma(a)| \cdot |O(w)|_a + |\sigma(b)| \cdot |O(w)|_b + |\sigma(c)| \cdot |O(w)|_c = |\sigma(O(w))|$. Per acabar, $O(\sigma(L)) = O(\{\sigma(w) \mid w \in L\}) = \{O(\sigma(w)) \mid w \in L\} = \{\sigma(O(w)) \mid w \in L\} = \sigma(\{O(w) \mid w \in L\}) = \sigma(O(L))$.**

(b) **Els mots w de L compleixen $w[i] = w[|w| - i + 1]$. Si $|w|$ es parell, això implica que cada posició queda univocament associada a una altra diferent, de manera que ambdues contenen el mateix símbol. En tal cas, $O(w)$ ha de ser de la forma $a^i b^j$ on $i, j \in \dot{2}$. En cas contrari, si $|w|$ és senar, tenim la mateixa relació biunívoca excepte pel símbol d'enmig. En tal cas, $O(w)$ ha de ser de la forma $a^i b^j$ amb o bé $i \in \dot{2},\ j \in \dot{2}+1$ o bé $i \in \dot{2}+1,\ j \in \dot{2}$. En general, qualsevol mot w de L compleix que $O(w)$ és de la forma $a^i b^j$ on $i \in \dot{2} \ \vee \ j \in \dot{2}$. Per a la direcció contrària, si considerem qualsevol mot de la forma $v = a^i b^j$ on $i \in \dot{2} \ \vee \ j \in \dot{2}$, no és difícil veure que existeix $w \in L$ tal que $O(w) = v$: si $i \in \dot{2}$ llavors només cal que agafem $w = a^{i/2} b^j a^{i/2}$, i si $j \in \dot{2}$ llavors només cal que agafem $w = b^{j/2} a^i b^{j/2}$. En conclusió, $O(L) = \{a^i b^j \mid i \in \dot{2} \ \vee \ j \in \dot{2}\}$. És fàcil d'obtenir directament el DFA mínim per a $O(L)$. Al principi, mentre llegim a's, només cal que recordem si n'hem vist un nombre parell o senar fins al moment. A partir de l'arribada de la primera b, només cal que anem recordant si n'hem vist un nombre parell o senar fins al moment. En aquest punt, l'arribada d'una nova a ens fa saltar a un estat pou.**



(c) **La condició que defineix L garanteix que qualsevol mot w de L compleix $|w|_a \geqslant |w|_b + |w|_c$. per tant, $O(w)$ serà de la forma $a^i b^j c^k$ amb $i \geqslant j + k$. Per altra banda, per a qualsevol mot v de la forma $a^i b^j c^k$ amb $i \geqslant j + k$ tenim que $w = (ab)^j (ac)^k a^{i-j-k}$ compleix $w \in L$ i $O(w) = v$. Així doncs, $O(L) = \{a^i b^j c^k \mid i \geqslant j + k\}$. Una CFG no ambigua per a aquesta descripció s'obté fàcilment:**

$$S \to aSc \mid X$$
$$X \to aXb \mid Y$$
$$Y \to aY \mid \lambda$$

2. (3.5 punts en total)

   (a) (0.75 punts, en blanc 0.1, dificultat estimada: mitjana/baixa, Temps de resolució estimat: 15 minuts) Demostreu que la implicació "L decidible $\Rightarrow O(L)$ decidible" és certa per a qualsevol llenguatge L.

   (b) (0.75 punts, en blanc 0.1, dificultat estimada: mitjana/baixa, Temps de resolució estimat: 15 minuts) Demostreu que la implicació "L semi-decidible $\Rightarrow O(L)$ semi-decidible" és certa per a qualsevol llenguatge L.

   (c) (1 punt, en blanc 0.1, dificultat estimada: mitjana, Temps de resolució estimat: 15 minuts) Demostreu **amb un contraexemple** que la implicació "L regular $\Rightarrow O(L)$ incontextual" **no** és certa per a alguns llenguatges L.

   (d) (1 punts, en blanc 0.1, dificultat estimada: alta, Temps de resolució estimat: 20 minuts) Demostreu **amb un contraexemple** que la implicació "$O(L)$ decidible $\Rightarrow L$ decidible" **no** és certa per a alguns llenguatges L.

Resposta:

(a) Sigui L un llenguatge decidible. Llavors existeix una màquina M_L que decideix L. En conseqüència, la següent màquina decideix $O(L)$:

Entrada v
  Per cada w **tal que** $|w| = |v|$ **fer:**
    Si $O(w) = v$ **i** $M_L(w)$ **accepta llavors Acceptar**
  Rebutjar

Notem que la màquina para amb tota entrada perquè el bucle considera un nombre finit de casos i M_L para amb tota entrada. A més a més, és clar que reconeix $O(L) = \{O(w) \mid w \in L\}$.

(b) Sigui L un llenguatge semi-decidible. Llavors existeix una màquina M_L que semi-decideix L. En conseqüència, la següent màquina semi-decideix $O(L)$:

Entrada v
  t**:=1**
  Executar indefinidament:
    Per cada w **tal que** $|w| = v$ **fer:**
      Si $O(w) = v$ **i** $M_L(w)$ **accepta en** t **passos llavors Acceptar**
    t**++**

Si una entrada v és de $O(L)$, llavors existeix w tal que $O(w) = v$ i $w \in L$. Per tant, $M_L(w)$ accepta en un cert nombre de passos T. Quan la variable t de la màquina anterior arribi a valer T, es considerarà w i la màquina acceptarà.

En el cas d'una entrada v que no sigui de $O(L)$, la màquina anterior es penja, doncs no es trobarà mai un cas de w on $O(w) = v$ i tal que $M_L(w)$ accepta. Això conclou que la màquina anterior semi-decideix $O(L)$.

(c) **Considerem el llenguatge regular L representat per l'expressió regular $L = (abc)^*$. Resulta que $O(L) = \{a^n b^n c^n \mid n \geqslant 0\}$, que és un conegut llenguatge no incontextual. Això ens dona el contraexemple que buscàvem.**

(d) **Considerem el llenguatge $L = a^+b^* \cup \{baa^x \mid x \in K\}$. L no és decidible, perquè la transformació computable i total $x \mapsto baa^x$ ens redueix K a L. Però resulta que $O(L) = a^+b^*$. Això ens dona el contraexemple que buscàvem.**

3. (1.25 punts, en blanc 0.15, dificultat estimada: mitjana, Temps de resolució estimat: 20 minuts) Classifiqueu raonadament com a decidible, semi-decidible però no decidible, o no semi-decidible, el següent conjunt, el qual conté les parelles de programes tals que els mots acceptats alhora per tots dos són justament els mots ordenats de $\{a, b, c\}^*$:

$$L = \{\langle p, q\rangle \mid (\mathcal{L}_p \cap \mathcal{L}_q) = \{w \in \{a, b, c\}^* \mid w = O(w)\}\}$$

Tingueu en compte que, tot i que els programes reben nombres naturals com a entrada, podeu assumir que aquests naturals codifiquen mots sobre $\{a, b, c\}$ de forma biunívoca. Per tant, podeu assumir que l'entrada d'un programa és un mot sobre $\{a, b, c\}$.

Resposta: reduïm $\bar{K}$ a L per a demostrar que L no és semi-decidible:

$$x \mapsto \left\langle p = \left[\begin{array}{l}\textbf{Entrada } w\\ \quad \textbf{Si } O(w) = w \textbf{ llavors Acceptar}\\ \quad \textbf{Executar } M_x(x)\\ \quad \textbf{Acceptar}\end{array}\right], q = \left[\begin{array}{l}\textbf{Entrada } w\\ \quad \textbf{Si } O(w) = w \textbf{ llavors Acceptar}\\ \quad \textbf{Executar } M_x(x)\\ \quad \textbf{Acceptar}\end{array}\right]\right\rangle$$

És evident que la transformació és computable i total. Es preserva la resposta afirmativa perquè, si x pertany a $\bar{K}$, llavors $M_x(x)\uparrow$, i per tant els programes M_p, M_q només accepten les entrades que cumpleixen $O(w) = w$, ja que en cas contrari es penjen, de manera que $\mathcal{L}_p = \mathcal{L}_q = \{w \in \{a, b, c\}^* \mid w = O(w)\}$. Es preserva la resposta negativa perquè, si x no pertany a $\bar{K}$, llavors $M_x(x)\downarrow$, i per tant els programes M_p, M_q accepten qualsevol entrada, de manera que $\mathcal{L}_p = \mathcal{L}_q = \{a, b, c\}^* \neq \{w \in \{a, b, c\}^* \mid w = O(w)\}$.

4. (2.25 punts en total) Considerem els següents problemes/conjunts, on s'assumeix que ni u ni v ni cap costat de regla de R és el mot buit (de fet s'assumeix que la pròpia codificació de l'entrada exclou aquests casos):

`Accessibilitat`$_{ab}$ (decidir si un mot u pot arribar a un mot v aplicant regles de reescriptura d'un conjunt R, on s'assumeix implícitament que tots els mots es defineixen sobre $\{a, b\}$):

$$A_{ab} = \{\langle u, v, R\rangle \mid u \to_R^* v\}$$

`Accessibilitat-ordenant-inici`$_{abc}$ (decidir si la ordenació d'un mot u pot arribar a un mot v a base d'aplicar regles de reescriptura d'un conjunt R, on s'assumeix implícitament que tots els mots es defineixen sobre $\{a, b, c\}$):

$$A_{abc,O(\texttt{inici})} = \{\langle u, v, R\rangle \mid O(u) \to_R^* v\}$$

`Accessibilitat-ordenat`$_{abc}$ (decidir si un mot u pot arribar a algun mot ordenat a base d'aplicar regles de reescriptura d'un conjunt R, on s'assumeix implícitament que tots els mots es defineixen sobre $\{a, b, c\}$):

$$A_{abc,\texttt{ordenat}} = \{\langle u, R\rangle \mid \exists v : u \to_R^* O(v)\}$$

(a) (1 punt, en blanc 0.15, dificultat estimada: mitjana, Temps de resolució estimat: 20 minuts) Demostreu raonadament que $A_{abc,O(\texttt{inici})}$ és indecidible amb una reducció des d'A_{ab}.

(b) (1.25 punts, en blanc 0.15, dificultat estimada: alta, Temps de resolució estimat: 20 minuts) Demostreu raonadament que $A_{abc,\texttt{ordenat}}$ és indecidible amb una reducció des d'A_{ab}.

Fixeu-vos que A_{ab} es defineix sobre $\{a,b\}$, i que $A_{abc,O(\texttt{inici})}$ i $A_{abc,\texttt{ordenat}}$ es defineixen sobre $\{a,b,c\}$. Per tant, si us plantegeu afegir símbols nous en les reduccions, i que aquestes reduccions siguin senzilles, el més raonable seria afegir només c's.

Resposta:

(a) Proposem com a reducció la transformació següent:

$$\langle u,v,R\rangle \;\mapsto\; \langle c,v,R'=R\cup\{c\to u\}\rangle$$

És clar que la transformació és computable i total. Veiem que preserva la resposta afirmativa suposant $u\to_R^* v$. Llavors, tenim $O(c)=c\to_{R'} u\to_R^* v$, i per tant $O(c)\to_{R'}^* v$. Això conclou la demostració de preservació de la resposta afirmativa.

Ara, veiem que es preserva la resposta negativa suposant $O(c)\to_{R'}^* v$. Donat que $O(c)=c$, i que la única regla on c està involucrada és $c\to u$ (degut a que R es defineix sobre $\{a,b\}$), resulta que la derivació anterior és de la forma $c\to_{c\to u} u\to_{R'}^* v$. En la subderivació $u\to_{R'}^* v$ no es pot haver utilitzat la regla $c\to u$, ja que u,v,R es defineixen sobre $\{a,b\}$. Per tant, la subderivació anterior és de la forma $u\to_R^* v$. Això conclou la demostració de preservació de la resposta negativa.

(b) Proposem com a reducció la transformació següent:

$$\langle u,v,R\rangle \;\mapsto\; \langle cuc,R'=R\cup\{cvc\to c\}\rangle$$

És clar que la transformació és computable i total. Veiem que preserva la resposta afirmativa suposant $u\to_R^* v$. Llavors, tenim $cuc\to_R^* cvc\to_{cvc\to c}^* c$, i com que $O(c)=c$, això conclou la demostració de preservació de la resposta afirmativa.

Ara, veiem que es preserva la resposta negativa suposant que existeixen x,w tals que $O(x)=w$ i $cuc\to_{R'}^* w$. Donat que u,R es defineixen sobre $\{a,b\}$, si en aquesta derivació no es fes servir la regla $cvc\to c$, llavors w seria de la forma $cw'c$ amb $w'\in\{a,b\}^+$, i no es compliria $O(x)=w$. Per tant, $cvc\to c$ s'ha fet servir almenys un cop. Pel que acabem de comentar, just abans de fer servir $cvc\to c$, s'ha accedir a un mot de la forma $cw'c$ amb $w'\in\{a,b\}^+$. Però llavors, aquest w' ha de ser precisament v. Així doncs, tenim una subderivació $cuc\to_R^* cvc$. Tal cosa implica $u\to_R^* v$, i això conclou la demostració de preservació de la resposta negativa.