

# Teoria de la Computació QP-2017

Exercicis per a l'avaluació contínua

## 1 Teoria de Llenguatges

[Vídeos 1, 2 i 3]

1. Formalitzeu els següents llenguatges utilitzant la notació clàssica de conjunts, com a parell variable de mot i propietat sobre aquest. Feu servir quantificadors universals i existencials, operadors booleans i les notacions sobre mides de mots que hem introduït.
  - (a) Llenguatge dels mots sobre  $\{a, b\}$  que contenen el submot  $ab$ .
  - (b) Llenguatge dels mots sobre  $\{a, b\}$  tals que a la dreta de cada submot  $ab$  hi ha algun submot  $ba$ .
  - (c) Llenguatge dels mots sobre  $\{a, b\}$  que contenen el submot  $ab$  i el submot  $ba$ .
  - (d) Llenguatge dels mots sobre  $\{a, b, c\}$  tals que entre cada dues  $b$ 's hi ha alguna  $a$ .
  - (e) Llenguatge de mots sobre  $\{a, b\}$  tal que tota ocurrència de  $b$  està en posició parell (el primer símbol d'un mot ocupa la posició 1).
  - (f) Llenguatge dels mots sobre  $\{a, b\}$  amb algun prefix amb més  $b$ 's o igual que  $a$ 's.
  - (g) Llenguatge dels mots sobre  $\{a, b\}$  tals que qualsevol prefixe seu té més  $b$ 's o igual que  $a$ 's.
  - (h) Llenguatge dels mots sobre  $\{a, b\}$  amb algun prefix de mida parell amb més  $b$ 's o igual que  $a$ 's.
  - (i) Llenguatge dels mots sobre  $\{a, b\}$  tals que qualsevol prefixe seu de mida parell té més  $b$ 's o igual que  $a$ 's.
  - (j) Llenguatge dels mots sobre  $\{a, b\}$  que tenen un prefix i un sufix idèntics de mida major que 0 i menor que la mida del propi mot.
2. Argumenteu si són certes (amb una justificació) o falses (amb un contraexemple) les següents afirmacions sobre mots  $x, y, z$  y llenguatges  $A, B, C$  en general.
  - (a)  $xy = yx$ .
  - (b)  $xy = xz \Rightarrow y = z$ .
  - (c)  $A(BC) = (AB)C$ .
  - (d)  $AB = BA$ .
  - (e)  $A \neq \emptyset \wedge AB = AC \Rightarrow B = C$ .
  - (f)  $A \neq \emptyset \wedge B \neq \emptyset \wedge AB = CD \wedge (\forall u \in A, v \in C : |u| = |v|) \Rightarrow A = C \wedge B = D$ .
  - (g)  $(A \cup B)C = AC \cup BC$  y  $A(B \cup C) = AB \cup AC$ .
  - (h)  $(A \cap B)C \subseteq AC \cap BC$  i  $A(B \cap C) \subseteq AB \cap AC$ .
  - (i)  $(A \cap B)C \supseteq AC \cap BC$  i  $A(B \cap C) \supseteq AB \cap AC$ .

3. Argumenteu si són certes (amb una justificació) o falses (amb un contraexemple) les següents afirmacions en general.

- (a)  $L^* = \{a, b\}^* \Rightarrow \{a, b\} \subseteq L$ .
- (b)  $L_1^* L_2^* \subseteq (L_1 L_2)^*$ .
- (c)  $L_1^* L_2^* \supseteq (L_1 L_2)^*$ .
- (d)  $L_1^+ \cup L_2^+ = \{a, b\}^+ \wedge L_1^+ \cap L_2^+ = \emptyset \Rightarrow L_1 = \emptyset \vee L_2 = \emptyset$ .
- (e)  $(L_1^* \cup L_2^*) \subseteq (L_1 \cup L_2)^*$ .
- (f)  $(L_1^* \cup L_2^*) \supseteq (L_1 \cup L_2)^*$ .
- (g)  $(L_1 \cap L_2)^* \subseteq (L_1^* \cap L_2^*)$ .
- (h)  $(L_1 \cap L_2)^* \supseteq (L_1^* \cap L_2^*)$ .
- (i)  $L_1^* \subseteq L_2^* \Rightarrow L_1 \subseteq L_2$ .
- (j)  $\overline{L^*} \subseteq \overline{L} \subseteq \overline{L^*}$ .
- (k)  $\overline{L^*} \supseteq \overline{L} \supseteq \overline{L^*}$ .
- (l)  $L_1 \neq \emptyset \wedge L_2 \neq \emptyset \wedge L_1 \cap L_2 = \emptyset \Rightarrow L_1^* \neq L_2^*$ .
- (m)  $L^+ L^+ \subseteq L^+$ .
- (n)  $L^+ L^+ \supseteq L^+$ .
- (o)  $(L^2)^* \subseteq (L^*)^2$ .
- (p)  $(L^2)^* \supseteq (L^*)^2$ .
- (q)  $L \subseteq L^2 \Leftrightarrow (\lambda \in L) \vee (L = \emptyset)$ .
- (r)  $L^2 \subseteq L \Leftrightarrow L = L^+$ .
- (s)  $(\lambda \in L) \wedge (L^2 \subseteq L) \Leftrightarrow L = L^*$ .
- (t)  $L = L^2 \Rightarrow (L = L^*) \vee (L = \emptyset)$ .

4. Argumenteu si són certes (amb una justificació) o falses (amb un contraexemple) les següents afirmacions en general.

- (a)  $(xy)^R = y^R x^R$ .
- (b)  $(L_1 L_2)^R = L_2^R L_1^R$ .
- (c)  $(L_1 \cup L_2)^R = L_1^R \cup L_2^R$ .
- (d)  $(L_1 \cap L_2)^R = L_1^R \cap L_2^R$ .
- (e)  $\overline{L^R} = \overline{L}^R$ .
- (f)  $(L^*)^R = (L^R)^*$ .
- (g)  $(L_1 L_2)^R = L_1^R L_2^R \Rightarrow L_1 = L_2$ .

5. Quines de les següents definicions de la funció  $\sigma$  defineixen un morfisme (és a dir, compleixen  $\sigma(xy) = \sigma(x)\sigma(y)$  per a mots  $x, y$  qualssevol).

- (a)  $\sigma(a_1 a_2 \cdots a_n) = a_1 a_1 a_2 a_2 \cdots a_n a_n$ .

- (b)  $\sigma(a_1 a_2 a_3 \cdots a_n) = a_1 a_2 a_2 a_3 a_3 a_3 \cdots \overbrace{a_n \cdots a_n}^n$ .
- (c)  $\sigma(w) = w$ .
- (d)  $\sigma(w) = \lambda$ .
- (e)  $\sigma(w) = a^{|w|}$ .
- (f)  $\sigma(w) = w^R$ .
- (g)  $\sigma(w) = \sigma_1(\sigma_2(w))$  per a morfismes  $\sigma_1, \sigma_2$ .
6. Argumenteu si són certes (amb una justificació) o falses (amb un contraexemple) les següents afirmacions en general, on  $\sigma$  és un morfisme.
- (a)  $\sigma(L_1 L_2) = \sigma(L_1) \sigma(L_2)$ .
- (b)  $\sigma(L^n) = \sigma(L)^n$ .
- (c)  $\sigma(L_1 \cup L_2) = \sigma(L_1) \cup \sigma(L_2)$ .
- (d)  $\sigma(L^*) = \sigma(L)^*$ .
- (e)  $\sigma(L^R) = \sigma(L)^R$ .
- (f)  $\sigma(\overline{L}) = \overline{\sigma(L)}$ .
- (g)  $\sigma(L) = L \Rightarrow \forall x \in L : \sigma(x) = x$ .
7. Argumenteu si són certes (amb una justificació) o falses (amb un contraexemple) les següents afirmacions en general.
- (a)  $|L_1| \cdot |L_2| = |L_1 \cdot L_2|$ .
- (b)  $(\forall a, b \in \Sigma : (a \neq b \Rightarrow \sigma(a) \neq \sigma(b))) \Rightarrow |\sigma(L)| = |L|$ , on  $\sigma$  és un morfisme.
- (c)  $|L^R| = |L|$ .
- (d)  $|L^n| = |L|^n$ .
8. Donat un llenguatge  $L$ , *shiftar*  $L$  dona 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, formalment:  $S(L) = \{vu \mid uv \in L\}$ . Argumenteu si són certes (amb una justificació) o falses (amb un contraexemple) les següents afirmacions en general.
- (a)  $S(L)^* \subseteq S(L^*)$ .
- (b)  $S(L)^* \supseteq S(L^*)$ .
- (c)  $\overline{S(L)} = S(\overline{L})$ .
- (d)  $S(L^R) = S(L)^R$ .
- (e)  $S(L_1 \cup L_2) = S(L_1) \cup S(L_2)$ .
- (f)  $S(L_1 \cap L_2) = S(L_1) \cap S(L_2)$ .
- (g)  $S(L_1 L_2) = S(L_1) S(L_2)$ .
- (h)  $S(\sigma(L)) = \sigma(S(L))$ , on  $\sigma$  és un morfisme.
9. Demostrad que no hay ninguna palabra  $w$  que cumpla  $aw = wb$ .
10. Demostrad que, para cualquier alfabeto  $\Sigma$ , hay un único lenguaje  $L$  que cumple  $L = \overline{\Sigma L}$ .  
Cuál es este lenguaje?

## 2 Autòmats finits

[Vídeos del 4 al 13]

1. Obtén los DFA's mínimos  $A_1, A_2$  para  $L_1 = \{xaay \mid x, y \in \{a, b\}^*\}$  y  $L_2 = \{xbby \mid x, y \in \{a, b\}^*\}$ , respectivamente, y calcula explícitamente el DFA intersección  $A_1 \cap A_2$ .
2. Obtén los DFA's mínimos  $A_1, A_2$  para  $L_1 = \{xaay \mid x, y \in \{a, b\}^*\}$  y  $L_2 = \{xbby \mid x, y \in \{a, b\}^*\}$ , respectivamente, y calcula explícitamente el DFA unión  $A_1 \cup A_2$ .
3. Obtén los DFA's mínimos  $A_1, A_2$  para  $L_1 = \{xay \mid x, y \in \{a, b\}^*\}$  y  $L_2 = \{xbby \mid x, y \in \{a, b\}^*\}$ , respectivamente, y calcula explícitamente el DFA unión  $A_1 \cup A_2$ .
4. Obtén el DFA mínimo  $A$  para  $L = \{aaw \mid w \in \{a, b\}^*\}$ , y calcula explícitamente  $\bar{A}$ .
5. Obtén el DFA mínimo  $A$  para  $L = \{w \in \{0, 1\}^* \mid \text{valor}_2(w) \in \dot{3}\}$ , y calcula explícitamente  $\bar{A}$ .
6. Obtén el DFA mínimo  $A$  para  $L = \{w \in \{0, 1\}^* \mid |w| \in \dot{3} + 1\}$ , y calcula explícitamente  $\bar{A}$ .
7. Obtén los DFA's mínimos  $A_1, A_2$  para  $L_1 = \{xaya \mid x, y \in \{a, b\}^*\}$  y  $L_2 = \{xbby \mid x, y \in \{a, b\}^*\}$ , respectivamente, y calcula explícitamente el  $\lambda$ -NFA concatenación  $A_1 \cdot A_2$ , determinízalo y minimízalo.
8. Obtén los DFA's mínimos  $A_1, A_2$  para  $L_1 = \{xaay \mid x, y \in \{a, b\}^*\}$  y  $L_2 = \{bxb \mid x \in \{a, b\}^*\}$ , respectivamente, y calcula explícitamente el NFA concatenación  $A_1 \cdot A_2$ , determinízalo y minimízalo.
9. Obtén los DFA's mínimos  $A_1, A_2$  para  $L_1 = \{xaya \mid x, y \in \{a, b\}^*\}$  y  $L_2 = \{bxb \mid x \in \{a, b\}^*\}$ , respectivamente, y calcula explícitamente el NFA concatenación  $A_1 \cdot A_2$ , determinízalo y minimízalo.
10. Obtén el DFA mínimo  $A$  para  $L = \{xay \in \{a, b\}^* \mid |y| = 1\}$ , y calcula explícitamente el NFA estrella  $A^*$ , determinízalo y minimízalo.
11. Obtén el DFA mínimo  $A$  para  $L = \{xaby \in \{a, b\}^* \mid |y| = 1\}$ , y calcula explícitamente el NFA estrella  $A^*$ , determinízalo y minimízalo.
12. Obtén el DFA mínimo  $A$  para  $L = \{axab \in \{a, b\}^* \mid |y| = 1\}$ , y calcula explícitamente el NFA estrella  $A^*$ , determinízalo y minimízalo.
13. Obtén el DFA mínimo  $A$  para  $L = \{w \in \{a, b\}^* \mid \forall w_1, w_2 (w = w_1aw_2 \Rightarrow |w_1|_b \in \dot{2})\}$ , y calcula explícitamente el NFA reverso  $A^R$ , determinízalo y minimízalo.
14. Obtén el DFA mínimo  $A$  para  $L = \{w \in \{a, b\}^* \mid \forall w_1, w_2 (w = w_1aw_2 \Rightarrow |w_1|_b \in \dot{2} + 1)\}$ , y calcula explícitamente el NFA reverso  $A^R$ , determinízalo y minimízalo.
15. Obtén el DFA mínimo  $A$  para  $L = \{w \in \{a, b\}^* \mid \forall w_1, w_2 (w = w_1aw_2 \Rightarrow |w_1| \in \dot{2})\}$ , y calcula explícitamente el NFA reverso  $A^R$ , determinízalo y minimízalo.

16. Obtén el DFA mínimo  $A$  para  $L = \{axbya \mid x, y \in \{a, b\}^*\}$ , y dado el morfismo definido por  $\sigma(a) = aa$ ,  $\sigma(b) = ba$ , calcula explícitamente el NFA imagen  $\sigma(A)$ , determinízalo y minimízalo.
17. Obtén el DFA mínimo  $A$  para  $L = \{axbyc \mid x, y \in \{a, b, c\}^*\}$ , y dado el morfismo definido por  $\sigma(a) = ab$ ,  $\sigma(b) = b$ ,  $\sigma(c) = \lambda$ , calcula explícitamente el NFA imagen  $\sigma(A)$ , determinízalo y minimízalo.
18. Obtén el DFA mínimo  $A$  para  $L = \{xbcy a \mid x, y \in \{a, b, c\}^*\}$ , y dado el morfismo definido por  $\sigma(a) = bbb$ ,  $\sigma(b) = a$ ,  $\sigma(c) = \lambda$ , calcula explícitamente el NFA imagen  $\sigma(A)$ , determinízalo y minimízalo.
19. Sea  $A$  el DFA mínimo que reconoce a los números binarios múltiplos de 3. Calcula  $\sigma^{-1}(A)$  tomando como  $\sigma$  los morfismos:
  - (a)  $\sigma(a) = 01$ ,  $\sigma(b) = 0$ ,  $\sigma(c) = \lambda$ .
  - (b)  $\sigma(a) = 10$ ,  $\sigma(b) = 0$ ,  $\sigma(c) = \lambda$ .
  - (c)  $\sigma(a) = 00$ ,  $\sigma(b) = 11$ ,  $\sigma(c) = \lambda$ .
  - (d)  $\sigma(a) = 001$ ,  $\sigma(b) = 101$ ,  $\sigma(c) = 0$ .
20. Diseña un algoritmo de coste razonable para encontrar los estados no accesibles de un DFA de entrada.
21. Diseña un algoritmo de coste razonable para decidir si un DFA de entrada acepta alguna palabra.
22. Diseña un algoritmo de coste razonable para decidir si un DFA de entrada acepta infinitas palabras.
23. Cuál es el coste del algoritmo de determinización de NFA's en DFA's.
24. Cuál es el coste temporal de las siguientes operaciones de DFA's:
  - (a) intersección.
  - (b) unión.
  - (c) complementario.
  - (d) concatenación (incluyendo determinización).
  - (e) estrella (incluyendo determinización).
  - (f) reverso (incluyendo determinización).
  - (g) aplicación de morfismo (incluyendo determinización).
  - (h) aplicación de morfismo inverso.
25. Por qué en nuestra definición de  $A^*$  añadimos un nuevo estado inicial, que es también aceptador, en el caso particular en que se cumpla  $\lambda \notin \mathcal{L}(A)$ ? En este caso, no bastaría con poner el propio estado inicial de  $A$  como aceptador?
26. Cuál es el coste del algoritmo de minimización con una implementación razonable.

27. Propón un algoritmo de coste razonable para saber si dos DFA's de entrada reconocen el mismo lenguaje.
28. Propón un algoritmo de coste razonable para saber, si dados dos DFA's de entrada, el lenguaje generado por el primero está incluido en el lenguaje generado por el segundo.
29. Justifiqueu la veracitat o falsetat de les següents afirmacions per a DFAs mínims  $A, A_1, A_2$ , NFAs  $B, B_1, B_2, B_3$  i morfisme  $\sigma$ . En cas que les operacions que apareixen hagin estat definides només per a DFAs, assumiu la seva extensió natural a NFAs.
  - (a)  $A_1 \cap A_2$  és DFA mínim.
  - (b)  $A_1 \cup A_2$  és DFA mínim.
  - (c)  $\bar{A}$  és DFA mínim.
  - (d)  $\sigma(A)$  és DFA.
  - (e)  $\sigma^{-1}(A)$  és DFA mínim.
  - (f)  $\bar{\bar{A}} = A$ .
  - (g)  $(B^R)^R = B$ .
  - (h)  $(B^*)^* = B^*$ .
  - (i)  $(B_1 B_2) B_3 = B_1 (B_2 B_3)$ .
  - (j)  $(B_1 B_2)^R = B_2^R B_1^R$ .
  - (k)  $(B^R)^* = (B^*)^R$ .
  - (l) En el cas en que  $A^R$  també sigui DFA, llavors podem concloure que és mínim.
30. Dado un lenguaje  $L$ , definimos  $\text{Prefijos}(L)$  como el lenguaje  $\{w|\exists x : (wx \in L)\}$ . Dado un DFA  $A$ , cómo se puede construir un DFA  $\text{Prefijos}(A)$  que cumpla  $\mathcal{L}(\text{Prefijos}(A)) = \text{Prefijos}(\mathcal{L}(A))$ .
31. Dado un lenguaje  $L$ , definimos  $\text{Sufijos}(L)$  como el lenguaje  $\{w|\exists x : (xw \in L)\}$ . Dado un DFA  $A$ , cómo se puede construir un DFA  $\text{Sufijos}(A)$  que cumpla  $\mathcal{L}(\text{Sufijos}(A)) = \text{Sufijos}(\mathcal{L}(A))$ .
32. Dado un natural  $n$  definimos  $L_n = \{w \in \{0,1\}^* | \exists k : (\text{valor}_2(w) = k \cdot 2^n)\}$ . Justifica que cualquier  $L_n$  es regular. Cuantos estados tiene el DFA mínimo que reconoce  $L_n$ ?
33. Cuantos estados tiene el DFA mínimo que reconoce las palabras sobre  $\{a, b, c\}$  que contienen al menos 100 ocurrencias de cada uno de estos tres símbolos.
34. Decimos que un NFA es de aceptación única si para cada palabra aceptada existe una única ejecución aceptadora. Demuestra que, para un NFA de aceptación única  $A$ ,  $A^R$  es un NFA de aceptación única.

### 3 Expressions regulars

[Vídeos 19 i 20]

1. Trobeu expressions regulars que representin els següents llenguatges transformant un DFA en una expressió regular segons el mètode basat en el lema d'Arden.
  - (a) Mots sobre  $\{a, b\}$  amb un nombre parell de  $a$ 's.
  - (b) Mots sobre  $\{a, b\}$  amb o bé un nombre parell de  $a$ 's, o bé un nombre parell de  $b$ 's.
  - (c) Mots sobre  $\{a, b\}$  acabats en  $ababa$ .
  - (d) Mots sobre  $\{a, b\}$  que no contenen el submot  $aba$ .
  - (e) Mots sobre  $\{a, b, c\}$  tals que, entre cada dues  $a$ 's hi ha almenys una  $b$ .
  - (f) Mots sobre  $\{0, 1\}$  amb almenys dos 0's consecutius.
2. Dada una expresión regular, cómo construirías otra, de forma sencilla, que genere el lenguaje reverso de la primera?
3. Dada una expresión regular  $r$  y un morfismo  $\sigma$ , cómo construirías otra expresión regular, de forma sencilla, que genere  $\sigma(\mathcal{L}(r))$ ?
4. Dadas dos expresiones regulares  $r_1$  y  $r_2$ , cómo decidirías:
  - (a)  $\mathcal{L}(r_1) = \mathcal{L}(r_2)$ .
  - (b)  $\mathcal{L}(r_1) \subseteq \mathcal{L}(r_2)$ .
  - (c)  $\mathcal{L}(r_1) = \emptyset$ .
  - (d)  $|\mathcal{L}(r_1)| = \infty$ .
  - (e)  $|\mathcal{L}(r_1) \cap \mathcal{L}(r_2)| = 0$ .
  - (f)  $|\mathcal{L}(r_1) \cap \mathcal{L}(r_2)| = \infty$ .
5. (Lema de Arden (bis)) Demuestra que  $BA^*$  es solución de la ecuación  $X = XA + B$ , que toda solución de esa ecuación contiene a  $BA^*$ , y que en el supuesto de que  $A$  no contenga  $\lambda$ , entonces  $BA^*$  es la única solución.
6. Aprovechando el resultado del ejercicio anterior, obtén una expresión regular para el complementario de las palabras que representan un múltiplo de 3 (es decir,  $\{w \in \{0, 1\}^* \mid \text{valor}_2(w) \in \dot{3}\}$ ). Para ello, escribe el autómata mínimo para ese lenguaje, crea una variable  $X_q$  para cada estado  $q$ , y crea ecuaciones con la idea de que la sustitución solución de cada  $X_q$  sea el lenguaje de las palabras que nos llevan desde el estado inicial al estado  $q$ .
7. Usa el método del ejercicio anterior para obtener expresiones regulares de los siguientes lenguajes. Compáralas con expresiones regulares que se obtienen aplicando el lema de Arden tal y como se explica en los vídeos.
  - (a) Mots sobre  $\{a, b\}$  amb un nombre parell de  $a$ 's.
  - (b) Mots sobre  $\{a, b\}$  amb o bé un nombre parell de  $a$ 's, o bé un nombre parell de  $b$ 's.

- (c) Mots sobre  $\{a, b\}$  acabats en  $ababa$ .
- (d) Mots sobre  $\{a, b\}$  que no contenen el submot  $aba$ .
- (e) Mots sobre  $\{a, b, c\}$  tals que, entre cada dues  $a$ 's hi ha almenys una  $b$ .
- (f) Mots sobre  $\{0, 1\}$  amb almenys dos 0's consecutius.

## 4 Gramáticas incontextuales

[Vídeos del 14 al 18]

1. Doneu una gramática no ambigua per a generar expressions amb operadores binaris de suma, resta, producte, divisió, i també admetent parentització explícita, de manera que l'arbre sintàctic generat es correspongui a la precedència habitual que donem als operadores.
2. Supón que una gramática cumple que en cada parte derecha hay como mucho una variable, y que los lenguajes generados desde dos partes derechas cualesquiera de una misma variable són disjuntos. Demuestra que, bajo estas condiciones, la gramática no es ambigua.

3. Justifica la ambigüetat o no ambigüetat de les següents CFG's:

(a)  $S \rightarrow (S)S|$

(b)  $S \rightarrow S(S)S|$

(c)

$$S \rightarrow aSb|B$$

$$B \rightarrow bAa|bCb|\lambda$$

$$A \rightarrow aAbA|bAaA|\lambda$$

$$C \rightarrow Aaa|aAa|aaA$$

(d)

$$S \rightarrow aU_1|aS|bZ_1|bS$$

$$Z_1 \rightarrow aU_2|bF$$

$$U_1 \rightarrow bU_2$$

$$U_2 \rightarrow bF|b$$

$$F \rightarrow aF|bF|a|b$$

(e)

$$S \rightarrow AaBA|ABaA|ACA|AbabA$$

$$B \rightarrow bb$$

$$C \rightarrow bB$$

$$A \rightarrow aA|bA|\lambda$$

(f)

$$S \rightarrow aU_1|aS|bZ_1|bS$$

$$Z_1 \rightarrow aU_2|bZ_2$$

$$U_1 \rightarrow bU_2$$

$$U_2 \rightarrow bF$$

$$Z_2 \rightarrow aF|bF$$

$$F \rightarrow aF|bF|\lambda$$



(g)

$$\begin{aligned} S &\rightarrow Z_1 a | Z_2 b \\ Z_1 &\rightarrow Z_1 a | U_1 b \\ Z_2 &\rightarrow U_2 a | Z_3 b \\ Z_3 &\rightarrow F a | U_2 \\ U_1 &\rightarrow U_2 | F b a \\ U_2 &\rightarrow F b \\ F &\rightarrow F a | F b | \lambda \end{aligned}$$

4. Muestra que la gramática unión  $G_1 \cup G_2$  de dos gramáticas no ambiguas  $G_1, G_2$  sí podría ser ambigua.
5. Muestra que la gramática concatenación  $G_1 \cdot G_2$  de dos gramáticas no ambiguas  $G_1, G_2$  sí podría ser ambigua.
6. Muestra que la gramática estrella  $G^*$  de una gramática no ambigua  $G$  sí podría ser ambigua.
7. Muestra que la gramática imagen  $\sigma(G)$  de una gramática no ambigua  $G$  por un morfismo  $\sigma$  sí podría ser ambigua.
8. Muestra que la gramática reverso  $G^R$  una gramática no ambigua  $G$  tampoco es ambigua.
9. Escribe el DFA mínimo para  $\{w \in \{a, b\}^* \mid |w|_a \in 2\}$ , y haz la intersección explícita de ese DFA con la CFG  $S \rightarrow aSa \mid bSb \mid a \mid b \mid \lambda$ .
10. Escribe el DFA mínimo para  $\{wa \mid w \in \{a, b\}^*\}$ , y haz la intersección explícita de ese DFA con la CFG  $S \rightarrow aSbS \mid bSaS \mid \lambda$ .
11. Escribe el DFA mínimo para  $\{aw \mid w \in \{a, b\}^*\}$ , y haz la intersección explícita de ese DFA con la CFG  $S \rightarrow aSbS \mid bSaS \mid \lambda$ .
12. Realizad la eliminación de  $\lambda$ -producciones, producciones unarias, y símbolos no-útiles, de las gramáticas siguientes:

(a)

$$S \rightarrow (S)S | \lambda$$

(b)

$$S \rightarrow SS | (S) | \lambda$$

(c)

$$\begin{aligned} S &\rightarrow AA \\ A &\rightarrow AA | \lambda \end{aligned}$$

(d)

$$\begin{aligned} S &\rightarrow A \\ A &\rightarrow B \\ B &\rightarrow c \end{aligned}$$

(e)

$$\begin{aligned} S &\rightarrow AB \\ A &\rightarrow a | \lambda \\ B &\rightarrow b | \lambda \end{aligned}$$

(f)

$$\begin{aligned} S &\rightarrow AB \\ A &\rightarrow aAb|\lambda \\ B &\rightarrow bBc|\lambda \end{aligned}$$

(g)

$$\begin{aligned} S &\rightarrow BC|\lambda \\ A &\rightarrow aA|\lambda \\ B &\rightarrow bB \\ C &\rightarrow c \end{aligned}$$

(h)

$$\begin{aligned} S &\rightarrow X|Y \\ X &\rightarrow Xc|A \\ A &\rightarrow aAb|\lambda \\ Y &\rightarrow aY|B \\ B &\rightarrow bBc|\lambda \end{aligned}$$

(i)

$$\begin{aligned} S &\rightarrow A|B|C \\ A &\rightarrow SaSbS|\lambda \\ B &\rightarrow SbSaS|\lambda \\ C &\rightarrow Cc|\lambda \end{aligned}$$

13. Sea  $G$  una CFG y sea  $C$  su conjunto de variables no-accesibles. Sea  $G'$  la gramática obtenida al borrar de  $G$  las variables de  $C$  junto con las reglas donde aparecen. Demuestra que el conjunto de variables accesibles de  $G'$  es, de hecho, el conjunto de todas las variables de  $G'$ .
14. Sea  $G$  una CFG y sea  $C$  su conjunto de símbolos no-fructíferos. Sea  $G'$  la gramática obtenida al borrar de  $G$  los símbolos de  $C$  junto con las reglas donde aparecen. Demuestra que el conjunto de símbolos fructíferos de  $G'$  es, de hecho, el conjunto de todas las variables de  $G'$ .
15. Sea  $G$  una CFG y sea  $C$  su conjunto de símbolos no-accesibles. Supongamos que todos los símbolos de  $G$  son fructíferos. Sea  $G'$  el resultado de borrar de  $G$  los símbolos de  $C$  junto con las reglas donde aparecen. Demuestra que el conjunto de símbolos fructíferos de  $G'$  es, de hecho, el conjunto de todas las variables de  $G'$ .
16. Da un ejemplo de gramática en la que, tras borrar un cierto símbolo útil, y todas las reglas en las que aparece, aún así el lenguaje generado se preserva.
17. Cuál es el coste temporal y espacial de eliminar las  $\lambda$ -producciones de una CFG.
18. Cuál es el coste temporal y espacial de eliminar las producciones unarias de una CFG.
19. Cuál es el coste temporal y espacial de eliminar los símbolos no útiles de una CFG.
20. Cuál es el coste temporal y espacial de pasar una CFG a CNF?
21. Cuál es el coste del algoritmo propuesto en el curso para decidir si una gramática genera una cierta palabra, si se realiza una implementación razonable del mismo? Cuál es el coste si se supone  $G$  fija, y que la entrada sólo contiene  $w$ ?

22. Sea  $n$  el número de pasos de derivación necesarios para generar una cierta palabra  $w$  con una cierta gramática  $G$  en forma normal de Chomsky. Podemos establecer alguna relación entre  $n$  y  $|w|$ ?
23. Justifiqueu la veracitat o falsetat de les següents afirmacions per a CFGs  $G, G_1, G_2, G_3$ .
- (a)  $(G^R)^R = G$ .
  - (b)  $(G_1 \cup G_2)^R = G_1^R \cup G_2^R$ .
  - (c)  $(G^R)^* = (G^*)^R$ .
  - (d)  $(G_1 \cup G_2)G = (G_1G) \cup (G_2G)$ .
  - (e)  $\sigma(G_1 \cup G_2) = \sigma(G_1) \cup \sigma(G_2)$ .
  - (f)  $G_1(G_2G_3) = (G_1G_2)G_3$ .
  - (g)  $(G_1G_2)^R = G_2^R G_1^R$ .
24. Propón un algoritmo de coste razonable para saber si una CFG de entrada genera alguna palabra.
25. Propón un algoritmo de coste razonable para saber si una CFG de entrada genera infinitas palabras.
26. Propón un algoritmo de coste razonable para saber si una CFG de entrada genera alguna palabra de tamaño par.
27. Propón un algoritmo de coste razonable para saber si una CFG de entrada genera infinitas palabras de tamaño par.
28. Propón un algoritmo de coste razonable para saber, dada una CFG  $G$  de entrada y un natural  $n$ , cuantos árboles de derivación distintos de palabras de tamaño  $n$  genera  $G$ .
29. Sigui  $L$  un llenguatge incontextual infinit. Demostra que hi ha una CFG  $G$  tal que  $\mathcal{L}(G) = L$  i totes les variables de  $G$  generen un llenguatge infinit.

## 5 No regularitat

[Vídeos 21 i 22]

1. Demostreu la no-regularitat dels següents llenguatges:

- (a)  $\{a^n b^n | n \in \dot{2}\}$ .
- (b)  $\{a^n b^n | n \in \dot{3}\}$ .
- (c)  $\{a^n b^m | n \neq m\}$ .
- (d)  $\{a^{2n} b^n | n \in \dot{2}\}$ .
- (e)  $\{w \in \{a, b\}^* | |w|_a = |w|_b\}$ .
- (f)  $\{a^n b^m | n \leq m\}$ .
- (g)  $\{a^n b^m | n \geq m\}$ .

- (h)  $\{c^m a^n b^n | (n, m \geq 0)\}$ .
  - (i)  $\{a, b\}^* \cup \{c^m a^n b^n | (m \geq 1) \wedge (n \geq 0)\}$ .
  - (j)  $\{w \in \{a, b\}^* | w = w^R\}$ .
  - (k)  $\{ww \in \{a, b\}^*\}$
  - (l)  $\{a^{n^2} | n \geq 0\}$ .
  - (m)  $\{a^{2^n} | n \geq 0\}$ .
  - (n)  $\{a^n | n \text{ apareix a la successió de Fibonacci}\}$ .
  - (o)  $\{a^n | n \text{ és primer}\}$ .
  - (p)  $\{a^n | n \text{ és parell o primer}\}$ .
  - (q)  $\{abab^2ab^3 \dots ab^n | n \geq 0\}$ .
  - (r)  $\{w_1 \# w_2 | w_1, w_2 \in \{0, 1\}^* \wedge (|w_1| < |w_2| \vee |w_1| \in \dot{2})\}$ .
  - (s)  $\{u \# v | u, v \in \{a, b\}^* \wedge v \text{ és submot de } u\}$ .
  - (t)  $\{w \in (a + b + c)^* \mid |w|_a \geq |w|_b \vee |w|_b \geq |w|_c\}$ .
  - (u) Qualsevol subconjunt infinit del llenguatge  $\{a^n b^n\}$ .
  - (v)  $\{w \in \{a, b\}^* \mid (|w| \in \dot{3} \Rightarrow |w|_a = |w|_b)\}$ .
  - (w)  $\{w_1 \# w_2 | w_1, w_2 \in \{0, 1\}^* \wedge \text{valor}_2(w_1) = \text{valor}_2(w_2)\}$ .
  - (x)  $\{w_1 \# w_2 | w_1, w_2 \in \{0, 1\}^* \wedge \text{valor}_2(w_1) = 1 + \text{valor}_2(w_2)\}$ .
  - (y)  $\{w_1 \# w_2 \# w_3 | w_1, w_2, w_3 \in \{0, 1\}^* \wedge \text{valor}_2(w_1) + \text{valor}_2(w_2) = \text{valor}_2(w_3)\}$ .
  - (z)  $\{xy \in \{a, b\}^* \mid |x|_a = 2|y|_b\}$
2. Considerem el llenguatge  $L_k = \{w \in (0+1)^* | \text{valor}_2(w) \leq k\}$ . Quins dels següents llenguatges són regulars per a qualsevol  $k$ :
- (a)  $L_k$ .
  - (b)  $\bigcup_{k \geq 1} L_k$ .
  - (c)  $\{w \# w | w \in L_k\}$ .
  - (d)  $\{1w \# 1w | 1w \in L_k\}$ .
  - (e)  $\{1w \# 1w | w \in L_k\}$ .
  - (f)  $\{w \# w | w \in L_k \wedge |w| \leq \text{valor}_2(w)\}$ .
  - (g)  $\{w \# w | 1w \in L_k\}$ .
  - (h)  $\{w_1 \# w_2 | w_1, w_2 \in L_k \wedge \text{valor}_2(w_1) = \text{valor}_2(w_2)\}$ .
  - (i)  $\{w_1 \# w_2 | \exists k : (w_1, w_2 \in L_k \wedge \text{valor}_2(w_1) = \text{valor}_2(w_2))\}$ .
3. Quins dels següents llenguatges podem assegurar que són no regulars sabent que  $A$  i  $B$  són no regulars i que  $\sigma$  és un morfisme.
- (a)  $\bar{A}$ .
  - (b)  $A \cup B$ .
  - (c)  $A \cap B$ .

- (d)  $A \cdot B$ .
- (e)  $A^R$ .
- (f)  $A^*$ .
- (g)  $S(A)$  (recordeu la definició de shiftar un llenguatge dels exercicis del primer tema).
- (h)  $\sigma(A)$ .
- (i)  $\sigma^{-1}(A)$ .

4. Determineu quin llenguatge genera cadascuna de les següents CFG's, i justifiqueu si aquest llenguatge és regular o no.

(a)

$$\begin{aligned} S &\rightarrow AB|CD \\ A &\rightarrow 0A0|0 \\ B &\rightarrow 1B1|\lambda \\ C &\rightarrow 0C0|\lambda \\ D &\rightarrow 1D1|\lambda \end{aligned}$$

(b)

$$\begin{aligned} S &\rightarrow aA|bB|\lambda \\ A &\rightarrow Sa|Sb \\ B &\rightarrow Sb \end{aligned}$$

(c)

$$\begin{aligned} S &\rightarrow AB \\ A &\rightarrow 0A0|1 \\ B &\rightarrow 1B1|0 \end{aligned}$$

(d)

$$S \rightarrow 0S0|0S1|\lambda$$

(e)

$$\begin{aligned} S &\rightarrow AB \\ A &\rightarrow 0A0|0A1|\lambda \\ B &\rightarrow 0B|1B|\lambda \end{aligned}$$

(f)

$$\begin{aligned} S &\rightarrow A|B \\ A &\rightarrow 0S0|1S1|\lambda \\ B &\rightarrow 0S1|1S0|\lambda \end{aligned}$$

(g)

$$\begin{aligned} S &\rightarrow A|B \\ A &\rightarrow 0A0|1A1|\lambda \\ B &\rightarrow 0B1|1B0|\lambda \end{aligned}$$

(h)

$$\begin{aligned} S &\rightarrow aSa|bSb|X \\ X &\rightarrow aXb|bXa|a|b|\lambda \end{aligned}$$

(i)

$$\begin{aligned} S &\rightarrow WXW' \\ X &\rightarrow aX|bX|\lambda \\ W &\rightarrow aW|bW|\lambda \\ W' &\rightarrow W'a|W'b|\lambda \end{aligned}$$

## 6 Autómatas con pila y jerarquía de Chomsky

[Vídeos del 23 al 26]

1. Muestra un ejemplo de lenguaje inambiguo que no sea DCFL.
2. Muestra un ejemplo de DCFL que no sea regular.
3. Muestra que los lenguajes inambiguos y los DCFL no son cerrados por morfismo directo.
4. Muestra que los lenguajes inambiguos y los DCFL no son cerrados por intersección.
5. Muestra que los DCFL no son cerrados por reverso.
6. Muestra que los lenguajes inambiguos y los DCFL no son cerrados por concatenación.
7. Muestra que los lenguajes inambiguos y los DCFL no son cerrados por estrella.
8. Sigui  $A$  regular,  $B$  CFL,  $C$  DCFL i  $\sigma$  un morfisme. Quins dels següents llenguatges podem assegurar que són regulars, quins podem assegurar que són DCFL, i quins podem assegurar que són CFL? Raoneu les respostes, i doneu contraexemples quan sigui necessari.

(a)  $\sigma^{-1}(B) \cap A$ .

(b)  $\overline{\sigma^{-1}(C)}$ .

(c)  $C^R$ .

(d)  $S(C)$  (recordeu la definició de shiftar un llenguatge donada en els problemes del primer tema).

(e)  $S(A)$ .

(f)  $(\overline{A} \cap \overline{C} \cup B)$ .

(g)  $(\sigma(B) \cap B)C$ .

(h)  $(\overline{A} - \sigma(B) \cap \overline{C})$ .

(i)  $(\sigma^{-1}(\sigma(B)) \cap B)C$ .

(j)  $\overline{(\sigma(A) \cap C)} \sigma(B \cap A)$ .

(k)  $\sigma(\sigma^{-1}(B) \cap A) \sigma^{-1}(C)$ .

9. Propón un algoritmo de coste razonable para saber, dado un DFA  $A$  y una CFG  $G$ , si se cumple  $\mathcal{L}(G) \subseteq \mathcal{L}(A)$ .
10. Propón un algoritmo de coste razonable para saber, dado un DFA  $A$  y una CFG  $G$ , si  $G$  genera infinitas palabras no aceptadas por  $A$ .
11. Propón un algoritmo de coste razonable para saber, dado un DFA  $A$  y una CFG  $G$ , si  $G$  genera alguna palabra de tamaño par no aceptada por  $A$ .

## 7 Máquinas de Turing

[Vídeos 27 i 28]

1. Escribid TM sencillas para los siguientes lenguajes:
  - (a)  $\{a^n b^n c^n | n \geq 0\}$ .
  - (b)  $\{w_1 \# w_2 | w_1, w_2 \in \{0, 1\}^* \wedge \text{valor}_2(w_1) = \text{valor}_2(w_2) + 1\}$ .
  - (c)  $\{ww | w \in \{0, 1\}^*\}$ .
  - (d)  $\{0^{2^n} | n \geq 0\}$
2. Escribid 2-TM (o 3-TM o 4-TM en caso de necesidad) sencillas para los siguientes lenguajes:
  - (a)  $\{a^n b^n c^n | n \geq 0\}$ .
  - (b)  $\{w_1 \# w_2 | w_1, w_2 \in \{0, 1\}^* \wedge \text{valor}_2(w_1) = \text{valor}_2(w_2) + 1\}$ .
  - (c)  $\{ww | w \in \{0, 1\}^*\}$ .
  - (d)  $\{0^{2^n} | n \geq 0\}$
  - (e)  $\{0^{n^2} | n \geq 0\}$
3. Argumenta a grandes rasgos que las máquinas de Turing no-deterministas no son más expresivas que las máquinas de Turing deterministas.
4. Considera el modelo de máquina que definimos a grandes rasgos así: una variante de los autómatas con pila donde, en vez de una pila, tenemos dos pilas, las transiciones dependen del contenido de la cima de ambas pilas, y en la acción de cada transición se puede o bien borrar el elemento de la cima o bien añadir nuevos elementos, todo ello en ambas pilas. Justifica a grandes rasgos que este modelo puede simular una máquina de Turing, y que, por tanto, es Turing-completo.
5. Considera el modelo de máquina que definimos a grandes rasgos así: una variante de los autómatas con pila donde, en vez de una pila, tenemos una cola, las transiciones dependen del contenido del inicio de la cola, y en la acción de cada transición se puede borrar el elemento del inicio, y también añadir nuevos elementos al final de la cola. Justifica a grandes rasgos que este modelo puede simular una máquina de Turing, y que, por tanto, es Turing-completo.

## 8 Decidibilidad, semi-decidibilidad, computabilidad

[Vídeos 29, 30 i 31]

1. Demuestra que los lenguajes decidibles son cerrados por las siguientes operaciones:
  - (a) Intersección.
  - (b) Complementario.
  - (c) Resta (de conjuntos).
  - (d) Reverso.

- (e) Concatenación.
  - (f) Estrella.
  - (g) Morfismo inverso.
  - (h) Shiftado.
2. Demuestra que los lenguajes decidibles no son cerrados por morfismo directo.
  3. Demuestra que los lenguajes semi-decidibles son cerrados por las siguientes operaciones:
    - (a) Intersección.
    - (b) Concatenación.
    - (c) Reverso.
    - (d) Estrella.
    - (e) Morfismo directo.
    - (f) Morfismo inverso.
    - (g) Shiftado.
  4. Demuestra que los siguientes conjuntos son semi-decidibles:
    - (a)  $\{\langle x, y \rangle \mid M_x(y) \downarrow\}$ .
    - (b)  $\{x \mid \exists y : M_x(y) \downarrow\}$ .
    - (c)  $\{\langle u, v, R \rangle \mid u \rightarrow_R^* v\}$ .
    - (d)  $\{G \in \mathbf{CFG} \mid G \text{ ambigua}\}$ .
    - (e)  $\{\langle G_1, G_2 \rangle \mid G_1, G_2 \in \mathbf{CFG} \wedge \mathcal{L}(G_1) \cap \mathcal{L}(G_2) \neq \emptyset\}$ .
  5. Sea  $B$  un conjunto semi-decidible y sea  $C$  un conjunto que cumple  $C = \{x \mid \exists y : \langle x, y \rangle \in B\}$ . Demuestra que  $C$  es semi-decidible.
  6. Sea  $C$  un conjunto infinito. Demuestra que  $C$  es decidible si y solo si existe una función computable, total, inyectiva y creciente cuya imagen es  $C$ .
  7. Sea  $C$  un conjunto infinito. Demuestra que  $C$  es semi-decidible si y solo si existe una función computable total e inyectiva cuya imagen es  $C$ .
  8. Sea  $f$  una función computable e inyectiva. Es  $f^{-1}$  computable e inyectiva?
  9. Sea  $f : \mathbb{N} \rightarrow \mathbb{N}$  una función estrictamente decreciente. Podemos asegurar que es computable?
  10. Sean  $A$  y  $B$  dos conjuntos tales que  $(A \cup B) - (A \cap B)$  es decidible y  $A$  es decidible. Eso implica que  $B$  es decidible?
  11. Sean  $A$  y  $B$  dos conjuntos tales que  $(A \cup B) - (A \cap B)$  es decidible y  $A$  es semi-decidible. Eso implica que  $B$  es semi-decidible?



## 9 No-decidibilidad, no-semi-decidibilidad, no-computabilidad

[Vídeos 32, 33 i 34]

1. Clasifica como decidibles, no decidibles pero semi-decidibles, o no semi-decidibles, los siguientes conjuntos.
  - (a)  $\{p \mid \mathcal{L}_p \text{ es finito}\}.$
  - (b)  $\{p \mid \mathcal{L}_p \text{ es infinito}\}.$
  - (c)  $\{p \mid M_p(p) = p\}.$
  - (d)  $\{p \mid \exists y : M_y(p) = p\}.$
  - (e)  $\{p \mid |\text{Dom}(\varphi_p)| \geq 10\}.$
  - (f)  $\{p \mid |\text{Dom}(\varphi_p)| \geq 0\}.$
  - (g)  $\{p \mid |\text{Im}(\varphi_p)| \geq 10\}.$
  - (h)  $\{p \mid |\text{Im}(\varphi_p)| \geq 0\}.$
  - (i)  $\{p \mid |\text{Im}(\varphi_p)| < |\text{Dom}(\varphi_p)| < \infty\}.$
  - (j)  $\{p \mid |\text{Dom}(\varphi_p)| < |\text{Im}(\varphi_p)| < \infty\}.$
  - (k)  $\{p \mid \varphi_p \text{ es inyectiva y total}\}.$
  - (l)  $\{p \mid \varphi_p \text{ es exhaustiva y total}\}.$
  - (m)  $\{p \mid \varphi_p \text{ es creciente y total}\}.$
  - (n)  $\{p \mid \varphi_p \text{ es total y estrictamente decreciente}\}.$
  - (o)  $\{p \mid \varphi_p \text{ es inyectiva parcial}\}.$
  - (p)  $\{p \mid \varphi_p \text{ es exhaustiva parcial}\}.$
  - (q)  $\{p \mid \varphi_p \text{ es creciente parcial}\}.$
  - (r)  $\{p \mid \varphi_p \text{ es estrictamente decreciente parcial}\}.$
2. Clasifica como decidibles, no decidibles pero semi-decidibles, o no semi-decidibles, los siguientes conjuntos.
  - (a)  $\{\langle p, q \rangle \mid \forall z : ((M_p(z) \downarrow \wedge M_q(z) \uparrow) \vee (M_p(z) \uparrow \wedge M_q(z) \downarrow))\}.$
  - (b)  $\{\langle p, z \rangle \mid \exists y : M_p(y) = z\}.$
  - (c)  $\{\langle p, z \rangle \mid \exists y : M_p(y) \neq z\}.$
  - (d)  $\{p \mid \mathcal{L}_p \text{ es incontextual}\}.$
  - (e)  $\{p \mid \mathcal{L}_p \text{ no es incontextual}\}.$
  - (f)  $\{p \mid \text{Dom}(\varphi_p) \in \text{Dec}\}.$
  - (g)  $\{p \mid \text{Dom}(\varphi_p) \notin \text{Dec}\}.$
  - (h)  $\{p \mid \text{Dom}(\varphi_p) \notin \text{semi-Dec}\}.$
  - (i)  $\{p \mid \text{Im}(\varphi_p) \in \text{Dec}\}.$
  - (j)  $\{p \mid \text{Im}(\varphi_p) \notin \text{Dec}\}.$
  - (k)  $\{p \mid \text{Im}(\varphi_p) \in \text{semi-Dec}\}.$

- (l)  $\{p | \text{Im}(\varphi_p) \notin \text{semi} - \text{Dec}\}.$
  - (m)  $\{p | p \leq 100 \wedge \text{Dom}(\varphi_p) \in \text{Dec}\}.$
  - (n)  $\{p | p \geq 100 \wedge \text{Dom}(\varphi_p) \in \text{semi} - \text{Dec}\}.$
  - (o)  $\{p | \forall y > p : \varphi_y \text{ es biyectiva}\}.$
  - (p)  $\{p | \forall y < p : \varphi_y \text{ es biyectiva}\}.$
  - (q)  $\{p | \exists y > p : \varphi_y \text{ es biyectiva}\}.$
  - (r)  $\{p | \exists y < p : \varphi_y \text{ es biyectiva}\}.$
  - (s)  $\{p | \exists y : \text{Dom}(\varphi_p) \subseteq \text{Dom}(\varphi_y)\}.$
  - (t)  $\{p | \exists y : \text{Dom}(\varphi_p) \supseteq \text{Dom}(\varphi_y)\}.$
  - (u)  $\{p | \text{Dom}(\varphi_p) \subseteq \dot{2}\}.$
  - (v)  $\{p | \text{Dom}(\varphi_p) \supseteq \dot{2}\}.$
3. Clasifica como decidibles, no decidibles pero semi-decidibles, o no semi-decidibles, los siguientes conjuntos.
- (a)  $K \times K.$
  - (b)  $\bar{K} \times K.$
  - (c)  $\bar{K} \times \bar{K}.$
  - (d)  $\overline{\bar{K} \times K}.$
  - (e)  $\{x \mid \text{el decimal } 3 \text{ aparece } x \text{ veces en el número } \pi\}.$
  - (f)  $\{\langle x, y \rangle \mid 0 \leq x \leq 9 \wedge \text{el decimal } x \text{ aparece } y \text{ veces consecutivas en la secuencia de decimales del número } \pi\}.$
4. Demuestra que  $K$  no se puede reducir a  $\bar{K}$ .
5. Demuestra que puede ocurrir que  $C$  sea decidible,  $f$  computable, y sin embargo  $f(C)$  no sea decidible.
6. Demuestra que puede ocurrir que  $C$  sea decidible,  $f$  computable y total, y sin embargo  $f(C)$  no sea decidible.
7. Demuestra que si  $C$  es semi-decidible y  $f$  es computable, entonces  $f(C)$  es semi-decidible.
8. Para cada una de las siguientes funciones indica si son computables, totales y cuál es su imagen.

$$(a) \ f(x) = \begin{cases} 1 & \text{si } \exists n : M_n(x) \downarrow \\ \uparrow & \text{si } \nexists n : M_n(x) \downarrow \end{cases}$$

$$(b) \ f(x) = \begin{cases} 1 & \text{si } \forall n : M_n(x) \downarrow \\ \uparrow & \text{si } \nexists n : M_n(x) \downarrow \end{cases}$$

$$(c) \ f(x) = \begin{cases} 1 & \text{si } \exists n : M_x(n) \downarrow \\ \uparrow & \text{si } \nexists n : M_x(n) \downarrow \end{cases}$$

$$(d) f(x) = \begin{cases} 1 & \text{si } \forall n : M_x(n) \downarrow \\ \uparrow & \text{si } \nexists n : M_x(n) \downarrow \end{cases}$$

9. La función característica de un conjunto  $C$  se define como:

$$\chi_C(x) = \begin{cases} 1 & \text{si } x \in C \\ 0 & \text{si } x \notin C \end{cases}$$

Demuestra que  $C$  es decidable si y solo si su función característica  $\chi_C$  es computable.

10. Justifica si los siguientes conjuntos de parejas son funciones, y si son funciones computables.

- (a)  $\varphi_3$ .
- (b)  $\{\langle x, y \rangle \mid M_x(x) = y\}$ .
- (c)  $\{\langle x, y \rangle \mid M_x(x) \leq y\}$ .
- (d)  $\{\langle x, y \rangle \mid M_x(x) \geq y\}$ .
- (e)  $\{\langle x, y \rangle \mid M_x(x) = M_y(y)\}$ .
- (f)  $\{\langle x, y \rangle \mid M_x(x) \text{ para en } y \text{ pasos o más}\}$ .
- (g)  $\{\langle x, y \rangle \mid M_x(x) \text{ para en exactamente } y \text{ pasos}\}$ .
- (h)  $\{\langle x, 1 \rangle \mid M_x(x) \downarrow\} \cup \{\langle x, 0 \rangle \mid M_x(x) \uparrow\}$ .
- (i)  $\{\langle x, 1 \rangle \mid M_x(x) \downarrow\}$ .
- (j)  $\{\langle x, 0 \rangle \mid M_x(x) \uparrow\}$ .
- (k)  $\{\langle x, y \rangle \mid y = |\{z \mid M_x(z) \downarrow\}|\}$ .

## 10 Problemas naturales indecibles

[Vídeos 35, 36 i 37]

1. Demostrad que los siguientes problemas son indecibles reduciendo desde accesibilidad de palabras.

- (a)  $\{\langle u, v, R \rangle \mid \exists w_1, w_2 : u \rightarrow_R^* w_1 v w_2\}$ .
- (b)  $\{\langle u, v, R \rangle \mid u \rightarrow_R^* v v\}$ .
- (c)  $\{\langle u, v, R \rangle \mid u u \rightarrow_R^* v\}$ .
- (d)  $\{\langle u, v, w, R \rangle \mid u \rightarrow_R^* v \text{ sin pasar por } w\}$ .
- (e)  $\{\langle u, v, R \rangle \mid u \rightarrow_R^* v \wedge \forall (l \rightarrow r) \in R : |l| = 1 \vee |r| = 1\}$ .
- (f)  $\{\langle u, v, R = \{l_1 \rightarrow r_1, \dots, l_n \rightarrow r_n\} \rangle \mid u \rightarrow_R^* v \wedge |l_1|, |r_1|, \dots, |l_n|, |r_n| \leq 2 \wedge |u| = |v| = 1\}$ .
- (g)  $\{\langle u, v, R \rangle \mid u \rightarrow_R^* v \text{ usando cada regla de } R \text{ al menos una vez}\}$ .

2. Demostrad que los siguientes problemas son indecibles reduciendo desde PCP o PCP-INI.

- (a)  $\{\langle u_1, v_1, \dots, u_n, v_n \rangle \mid n \in \dot{2} + 1 \wedge \exists 1 \leq i_1, \dots, i_k \leq n : (k > 0 \wedge u_{i_1} \cdots u_{i_k} = v_{i_1} \cdots v_{i_k})\}$ .
- (b)  $\{\langle u_1, v_1, \dots, u_n, v_n \rangle \mid \exists 1 \leq i_1, \dots, i_k \leq n : (k \in \dot{2} \wedge u_{i_1} \cdots u_{i_k} = v_{i_1} \cdots v_{i_k})\}$ .

- (c)  $\{\langle u_1, v_1, \dots, u_n, v_n \rangle \mid |u_1|, |v_1|, \dots, |u_n|, |v_n| \in \dot{2} \wedge \exists 1 \leq i_1, \dots, i_k \leq n : (k > 0 \wedge u_{i_1} \cdots u_{i_k} = v_{i_1} \cdots v_{i_k})\}$ .
- (d)  $\{\langle u_1, v_1, \dots, u_n, v_n \rangle \mid u_1, v_1, \dots, u_n, v_n \in \{0, 1\}^* \wedge \exists 1 \leq i_1, \dots, i_k \leq n : (k > 0 \wedge u_{i_1} \cdots u_{i_k} = v_{i_1} \cdots v_{i_k})\}$ .
3. Demostrad que los siguientes problemas son indecidibles reduciendo desde intersección vacía, o desde intersección no vacía.
- (a)  $\{\langle G_1, G_2 \rangle \mid |\mathcal{L}(G_1) \cap \mathcal{L}(G_2)| \geq 3\}$ .
- (b)  $\{\langle G_1, G_2 \rangle \mid |\mathcal{L}(G_1) \cap \mathcal{L}(G_2)| = 3\}$ .
- (c)  $\{\langle G_1, G_2 \rangle \mid |\mathcal{L}(G_1) \cap \mathcal{L}(G_2)| = \infty\}$ .
- (d)  $\{\langle G_1, G_2 \rangle \mid |\mathcal{L}(G_1)| = |\mathcal{L}(G_2)| = \infty \wedge |\mathcal{L}(G_1) \cap \mathcal{L}(G_2)| \neq \infty\}$ .
- (e)  $\{\langle G_1, G_2 \rangle \mid \Sigma_{G_1} = \Sigma_{G_2} = \{a, b\}^* \wedge \mathcal{L}(G_1) \cap \mathcal{L}(G_2) \neq \emptyset \wedge \mathcal{L}(G_1) \cap \mathcal{L}(G_2) \cap \{aa, bb\}^* = \emptyset\}$ .
- (f)  $\{\langle G_1, G_2 \rangle \mid \mathcal{L}(G_1) \cap \mathcal{L}(G_2) \neq \emptyset \wedge \nexists w \in (\mathcal{L}(G_1) \cap \mathcal{L}(G_2)) : |w| \in \dot{2} + 1\}$ .
- (g)  $\{\langle G_1, G_2 \rangle \mid \mathcal{L}(G_1) \cap \mathcal{L}(G_2) \neq \emptyset \wedge \nexists w \in (\mathcal{L}(G_1) \cap \mathcal{L}(G_2)) : |w| \in \dot{2}\}$ .
4. Demostrad que los siguientes problemas son indecidibles reduciendo desde no universalidad.
- (a)  $\{G \mid \mathcal{L}(G) \neq \Sigma^* - \{\lambda\}\}$ .
- (b)  $\{G \mid \mathcal{L}(G) \neq \Sigma^* - \{aab\}\}$ .
- (c)  $\{G \mid \mathcal{L}(G) \not\supseteq (aab\Sigma^*)\}$ .
- (d)  $\{G = \langle V, \Sigma, \delta, S \rangle \mid \Sigma = \{a, b\} \wedge \mathcal{L}(G) \neq \Sigma^*\}$ .
- (e)  $\{G = \langle V, \Sigma, \delta, S \rangle \mid \Sigma = \{a, b\} \wedge \mathcal{L}(G) \neq (aa + bb)^*\}$ .
- (f)  $\{G = \langle V, \Sigma, \delta, S \rangle \mid \Sigma = \{a, b\} \wedge \mathcal{L}(G) \neq (aa + bb + ab + ba)^*\}$ .
- (g)  $\{G = \langle V, \Sigma, \delta, S \rangle \mid \Sigma = \{a, b, c\} \wedge \mathcal{L}(G) \neq \Sigma^*\}$ .
- (h)  $\{G \mid |\overline{\mathcal{L}(G)}| = \infty\}$ .
5. Sea  $L$  un lenguaje regular. Demuestra que o bien  $\{G \mid \mathcal{L}(G) = L\}$  o bien  $\{G \mid \mathcal{L}(G) = \bar{L}\}$  es indecidible. Pista: Procede por reducción al absurdo suponiendo que ambos son decidibles y concluyendo que entonces universalidad es decidible.
6. Demuestra que los siguientes problemas son indecidibles reduciendo desde veracidad de fórmulas de lógica de primer orden interpretadas sobre palabras.
- (a) El problema de saber si una descripción de la forma  $\{w \mid F(w)\}$ , donde  $F$  es una fórmula de lógica de palabras con variable libre  $w$ , representa al lenguaje  $(a + b)^*$ .
- (b) El problema de saber si una descripción de la forma  $\{w \mid F(w)\}$ , donde  $F$  es una fórmula de lógica de palabras con variable libre  $w$ , representa al lenguaje vacío.
- (c) El problema de saber si una descripción de la forma  $\{w \mid F(w)\}$ , donde  $F$  es una fórmula de lógica de palabras con variable libre  $w$ , representa al lenguaje  $a^*b^*$ .
- (d) El problema de saber si una descripción de la forma  $\{w \mid F(w)\}$ , donde  $F$  es una fórmula de lógica de palabras con variable libre  $w$ , representa al lenguaje de las palabras palíndromas sobre  $\{a, b\}$ .

- (e) El problema de saber si una descripción de la forma  $\{w|F(w)\}$ , donde  $F$  es una fórmula de lógica de palabras con variable libre  $w$ , representa algún lenguaje finito.
- (f) El problema de saber si una descripción de la forma  $\{w|F(w)\}$ , donde  $F$  es una fórmula de lógica de palabras con variable libre  $w$ , representa algún lenguaje incontextual.
7. Demuestra que los siguientes problemas son decidibles:
- (a)  $\{\langle u, v, R = \{l_1 \rightarrow r_1, \dots, l_n \rightarrow r_n\} \rangle \mid u \rightarrow_R^* v \wedge \forall 1 \leq i \leq n : |l_i| \geq |r_i|\}$ .
  - (b)  $\{\langle u, v, R = \{l_1 \rightarrow r_1, \dots, l_n \rightarrow r_n\} \rangle \mid u \rightarrow_R^* v \wedge \forall 1 \leq i \leq n : |l_i| \leq |r_i|\}$ .
  - (c)  $\{\langle u, v, R = \{l_1 \rightarrow r_1, \dots, l_n \rightarrow r_n\} \rangle \mid u \rightarrow_R^* v \wedge |R| = 1\}$ .
  - (d)  $\{\langle u, v, R = \{l_1 \rightarrow r_1, \dots, l_n \rightarrow r_n\} \rangle \mid u \rightarrow_R^* v \wedge |u|, |v|, |l_1|, |r_1|, \dots, |l_n|, |r_n| \leq 10 \wedge \Sigma = \{0, 1\}\}$ .
  - (e)  $\{\langle u, v, R = \{l_1 \rightarrow r_1, \dots, l_n \rightarrow r_n\} \rangle \mid u \rightarrow_R^* v \wedge u, v, l_1, r_1, \dots, l_n, r_n \in a^*\}$ .
  - (f)  $\{\langle u_1, v_1, \dots, u_n, v_n \rangle \mid u_1, v_1, \dots, u_n, v_n \in a^* \exists 1 \leq i_1, \dots, i_k \leq n : u_{i_1} \dots u_{i_k} = v_{i_1} \dots v_{i_k}\}$ .
  - (g)  $\{\langle G, A \rangle \in \text{CFG} \times \text{DFA} \mid \mathcal{L}(G) \not\subseteq \mathcal{L}(A)\}$ .
8. Completa la siguiente reducción de PCP a intersección no vacía:  $\langle u_1, v_1, \dots, u_n, v_n \rangle \mapsto \langle S \rightarrow u_1 S v_1^R \mid \dots \mid u_n S v_n^R \mid u_1 \# v_1^R \mid \dots \mid u_n \# v_n^R, \dots? \dots \rangle$ .
9. Cuales de los siguientes problemas pueden ser decidibles y cuales pueden ser indecidibles:
- (a)  $\{\langle u, v \rangle \mid u \rightarrow_R^* v\}$  donde  $R$  es un sistema de reescritura fijado de antemano.
  - (b)  $\{R \mid u \rightarrow_R^* v\}$  donde  $u, v$  son palabras diferentes fijadas de antemano.
  - (c)  $\{\langle u_1, v_1, \dots, u_n, v_n \rangle \mid \exists 1 \leq i_1, \dots, i_k \leq n : u_{i_1} \dots u_{i_k} = v_{i_1} \dots v_{i_k}\}$  donde  $k$  es un número fijado de antemano.
  - (d)  $\{\langle u_1, v_1, \dots, u_n, v_n \rangle \mid \exists 1 \leq i_1, \dots, i_k \leq n : u_{i_1} \dots u_{i_k} = v_{i_1} \dots v_{i_k}\}$  donde  $n$  es un número fijado de antemano.
  - (e)  $\{G \mid \mathcal{L}(G) \supseteq \mathcal{L}(G')\}$  donde  $G'$  es una gramática fijada de antemano.
  - (f)  $\{G \mid \mathcal{L}(G) \not\subseteq \mathcal{L}(G')\}$  donde  $G'$  es una gramática fijada de antemano. Pista: piensa en la reducción de PCP a intersección no vacía del problema 8.
  - (g)  $\{G \mid \mathcal{L}(G) \cap \mathcal{L}(G') \neq \emptyset\}$  donde  $G'$  es una gramática fijada de antemano. Pista: piensa en la reducción de PCP a intersección no vacía del problema 8.
  - (h)  $\{G \in \text{CFG} \mid \mathcal{L}(G) \not\subseteq \mathcal{L}(A)\}$  donde  $A$  es un DFA fijado de antemano.
10. Dado un sistema de reescritura  $R = \{l_1 \rightarrow r_1, \dots, l_n \rightarrow r_n\}$  sobre el alfabeto  $\{a, b\}$ , llamaremos  $w_R$  a la palabra  $\#l_1\$r_1\#l_2\$r_2\# \dots \#l_n\$r_n\#$ , donde  $\#$  y  $\$$  son símbolos nuevos. Esencialmente,  $w_R$  es una palabra que codifica el sistema de reescritura. Demuestra que existe una fórmula de lógica de primer orden sobre palabras  $F(x, y, z)$ , donde  $x, y$  y  $z$  son variables libres, que cumple lo siguiente: dadas dos palabras  $u, v$  y un sistema de reescritura  $R$ , todos ellos sobre el alfabeto  $\{a, b\}$ , resulta que  $u \rightarrow_R^* v \Leftrightarrow F(u, v, w_R)$ .
11. Aprovechando el resultado del ejercicio anterior, Demuestra que los siguientes problemas son indecidibles reduciendo desde veracidad de fórmulas de lógica de primer orden interpretadas sobre palabras.

- (a) El problema de saber si una descripción de la forma  $\{w|F(w)\}$ , donde  $F$  es una fórmula de lógica de palabras con variable libre  $w$ , representa algún lenguaje decidable.
- (b) El problema de saber si una descripción de la forma  $\{w|F(w)\}$ , donde  $F$  es una fórmula de lógica de palabras con variable libre  $w$ , representa algún lenguaje semi-decible.
12. Justifica que  $\{G|\mathcal{L}(G) \text{ no es regular}\}$  es indecidible. Pista: retoca la reducción de PCP a intersección no vacía  $\langle u_1, v_1, \dots, u_n, v_n \rangle \mapsto \langle G_1, G_2 \rangle$  así:  $\langle u_1, v_1, \dots, u_n, v_n \rangle \mapsto \langle \{a^m b^m\} \cdot G_1, \{a^m b^m\} \cdot G_2 \rangle$ , donde  $a$  y  $b$  son símbolos nuevos. Haz lo mismo con la reducción de PCP a no-universalidad. Justifica que la gramática resultante, si no es universal, entonces el lenguaje que genera no es regular.