Teoria de la Computació

Tema 4: Gramàtiques incontextuals

Teoria:

- `Vídeos del 14 al 18`
- `Llibre TC Capítol 2 (Gramàtiques incontextuals) i capítol 3 (Normalització de gramàtiques)`

Exercicis per a l'avaluació contínua:

1. Doneu una gramàtica no ambígua per a generar expressions amb operadors binaris de suma, resta, producte, divisió, i també admetent parentització explícita, de manera que l'arbre sintàctic generat es correspongui a la precedéncia habitual que donem als operadors.

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)
   $$\begin{array}{rcl} S & \rightarrow & aSb|B \\ B & \rightarrow & bAa|bCb|\lambda \\ A & \rightarrow & aAbA|bAaA|\lambda \\ C & \rightarrow & Aaa|aAa|aaA \end{array}$$

   (d)
   $$\begin{array}{rcl} S & \rightarrow & aU_1|aS|bZ_1|bS \\ Z_1 & \rightarrow & aU_2|bF \\ U_1 & \rightarrow & bU_2 \\ U_2 & \rightarrow & bF|b \\ F & \rightarrow & aF|bF|a|b \end{array}$$

   (e)
   $$\begin{array}{rcl} S & \rightarrow & AaBA|ABaA|ACA|AbabA \\ B & \rightarrow & bb \\ C & \rightarrow & bB \\ A & \rightarrow & aA|bA|\lambda \end{array}$$

   (f)
   $$\begin{array}{rcl} S & \rightarrow & aU_1|aS|bZ_1|bS \\ Z_1 & \rightarrow & aU_2|bZ_2 \\ U_1 & \rightarrow & bU_2 \\ U_2 & \rightarrow & bF \\ Z_2 & \rightarrow & aF|bF \\ F & \rightarrow & aF|bF|\lambda \end{array}$$

(g)

$$\begin{array}{rcl} S & \to & Z_1a|Z_2b \\ Z_1 & \to & Z_1a|U_1b \\ Z_2 & \to & U_2a|Z_3b \\ Z_3 & \to & Fa|U_2 \\ U_1 & \to & U_2|Fba \\ U_2 & \to & Fb \\ F & \to & Fa|Fb|\lambda \end{array}$$

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 σ 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 $\{w \in \{a,b\}^* \mid |w|_a \in \dot{2}\}$, y haz la intersección explícita de ese DFA con la CFG $S \to aSa \mid bSb \mid a \mid b \mid \lambda$.

10. Escribe el DFA mínimo para $\{wa \mid w \in \{a,b\}^*\}$, y haz la intersección explícita de ese DFA con la CFG $S \to aSbS \mid bSaS \mid \lambda$.

11. Escribe el DFA mínimo para $\{aw \mid w \in \{a,b\}^*\}$, y haz la intersección explícita de ese DFA con la CFG $S \to aSbS \mid bSaS \mid \lambda$.

12. Realizad la eliminación de λ-producciones, producciones unarias, y símbolos no-útiles, de las gramáticas siguientes:

(a)

$$S \to (S)S|\lambda$$

(b)

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

(c)

$$\begin{array}{rcl} S & \to & AA \\ A & \to & AA|\lambda \end{array}$$

(d)

$$\begin{array}{rcl} S & \to & A \\ A & \to & B \\ B & \to & c \end{array}$$

(e)

$$\begin{array}{rcl} S & \to & AB \\ A & \to & a|\lambda \\ B & \to & b|\lambda \end{array}$$

(f)

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

(g)

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

(h)

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

(i)

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

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 λ-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.