Modelos de computación (2015-2016)

GRADO EN INGENIERÍA INFORMÁTICA

UNIVERSIDAD DE GRANADA

Práctica 1

José Carlos Martínez Velázquez

17 de octubre de 2015

1. Describir el lenguaje generado por las siguientes gramáticas en {a,b,c,d}*:

La metodología de trabajo de éste ejercicio será analizar las gramáticas de abajo a arriba, es decir, si tenemos $S, S_1, S_2, ..., S_n$, analizaremos primero $S_n, ..., S_2, S_1$ y aplicaremos todas las restricciones de las subgramáticas a S, que es el global.

a) $S \rightarrow aS_1b \quad ; \quad S_1 \rightarrow aS_1 \mid bS_1 \mid \varepsilon$

En primer lugar analizamos $S_1 \rightarrow aS_1 \mid bS_1 \mid \varepsilon$:

Esta gramática genera cualquier palabra que empiece por 'a', 'b' o sea ε, luego, en lenguaje formal:

$$\mathscr{L}(S_1) = \{a, b\}^*$$

Aplicamos esa restricción a $S \rightarrow aS_1b$

Se ve fácilmente que la gramática S genera cualquier palabra que empieza por 'a' y termina por 'b', pero no puede ser la palabra vacía. Por tanto:

$$\mathscr{L}(S) = \{u\epsilon\{a, b\}^* \ / \ u \ empieza \ por \ 'a' \ \bigwedge \ u \ termina \ por \ 'b'\}$$

b) $S \rightarrow aSa \mid bSb \mid S_1 \quad ; \quad S_1 \rightarrow a \mid b \mid \varepsilon$

En primer lugar analizamos $S_1 \rightarrow a \mid b \mid \varepsilon$:

Es sencillo ver que la gramática S_1 genera sólo y exclusivamente las palabras a, b y ε, por lo tanto:

$$\mathscr{L}(S_1) = \{u\epsilon\{a, b\}^* \ / \ u = a \ \bigvee \ u = b \ \bigvee u = \varepsilon\}$$

Aplicamos la restricción a $S \rightarrow aSa \mid bSb \mid S_1$

S va a generar palíndromos formados por los símbolos a y b. También podría generar la palabra vacía:

$$\mathscr{L}(S) = \{uxu^{-1} \ / \ u \ \epsilon \ \{a, b\}^* \ \bigwedge x \ \epsilon \ \{a, b, \varepsilon\}\}$$

c) $S \rightarrow aSb \mid aS_1b \quad ; \quad S_1 \rightarrow cS_1d \mid \varepsilon$

En primer lugar analizamos $S_1 \rightarrow cS_1d \mid \varepsilon$: Nos damos cuenta que la gramática S_1 genera cualquier sucesión de símbolos 'c' seguidos del mismo número de símbolos 'd', por lo tanto tenemos que:

$$\mathscr{L}(S_1) = \{c^id^i \ / \ i \geq 0\}$$

Aplicamos la restricción a $S \ \rightarrow \ aSb \mid aS_1b$

S genera cualquier sucesión de símbolos 'a' seguidos de cualquier palabra que genere S_1 seguida del mismo numero de símbolos 'b' que el numero de símbolos 'a', quizás así se vea más claro:

$$\mathscr{L}(S) = \{a^i c^j d^j b^i \ / \ i \geq 1, j \geq 0\}$$

d) $S \ \rightarrow \ S_1bbS_1 \quad ; \quad S_1 \ \rightarrow \ aS_1 \mid bS1 \mid \varepsilon$

En primer lugar analizamos $S_1 \ \rightarrow \ aS_1 \mid bS1 \mid \varepsilon$:

La gramática S_1 es capaz de crear cualquier palabra de $\{a, b\}^*$, por lo tanto:

$$\mathscr{L}(S_1) = \{a, b\}^*$$

Aplicamos la restricción a $S \ \rightarrow \ S_1bbS_1$ Con la gramatica S podemos generar cualquier palabra que tenga la subcadena 'bb', por tanto:

$$\mathscr{L}(S) = \{ubbw \ / \ u, w \ \epsilon \ \{a, b\}^*\}$$

2. Encontrar gramáticas de tipo 2 para los siguientes lenguajes sobre el alfabeto {a,b}. En cada caso determinar si los lenguajes generados son de tipo 3, estudiando si existe una gramática de tipo 3 que los genera.

La manera de proceder en éste ejercicio será contraria al ejercicio 1. Crearemos la "capa" superior S, definiendo "casos especiales", que serán los que gestionemos en las capas inferiores. De éste modo podremos encontrar la gramática de tipo 3 (si existe) directamente.

a) Palabras que tienen 2 o 3 b.

Creo la capa S. ¿Qué quiero que haga S? que genere palabras que empiecen por 'a' o por 'b'. Si empieza por a, puedo volver a generar cualquier letra, pero no puedo generar la palabra vacía hasta que no tenga al menos dos 'b'. Si genero una 'b', tengo que gestionar un caso especial, si genero otra 'b', otro caso especial en el que ya puedo generar la palabra vacía (cortar). Si genero una 'b' más, tengo que gestionar otro caso donde sólo pueda generar el símbolo 'a'. Entonces la gramática que creamos es:

$$\begin{aligned}
S &\rightarrow aS \mid bS_1 \\
S_1 &\rightarrow aS_1 \mid bS_2 \\
S_2 &\rightarrow aS_2 \mid bS_3 \mid \varepsilon \\
S_3 &\rightarrow aS_3 \mid \varepsilon
\end{aligned}$$

b) Palabras en las que el numero de b no es tres.
Aquí, hay que tener en cuenta que solo puedo generar la palabra vacía cuando el número de 'b' no sea tres, por tanto:

$$\begin{aligned} S &\rightarrow bS_1 \mid aS \mid \varepsilon \\ S_1 &\rightarrow bS_2 \mid aS_1 \mid \varepsilon \\ S_2 &\rightarrow bS_3 \mid aS_2 \\ S_3 &\rightarrow bS_3 \mid aS_3 \mid \varepsilon \end{aligned}$$

c) Palabras que no contienen la subcadena ab
Si genro una 'a', tengo que gestionar el caso especial de que sólo puedo generar más 'a' o la palabra vacía. Si genero una 'b', el siguiente símbolo puede ser cualquiera, luego:

$$\begin{aligned} S &\rightarrow aS_1 \mid bS \mid \varepsilon \\ S_1 &\rightarrow aS_1 \mid \varepsilon \end{aligned}$$

d) Palabras que no contienen la subcadena baa
Este tiene un poco más de miga. Si genero una 'a', puedo volver a generar cualquier símbolo. Si genero una 'b' tengo que gestionar un caso especial. Con una 'b' generada, puedo generar más 'b' sin problema, pero si genero una 'a' tengo que volver a gestionar otro caso especial. Con ba generado, sólo puedo generar 'b', lo que rompería la cadena prohibida y podría volver a generar, al menos, una 'a'. Veámoslo más claro:

$$\begin{aligned} S &\rightarrow aS \mid bS_1 \mid \varepsilon \\ S_1 &\rightarrow aS_2 \mid bS_1 \mid \varepsilon \\ S_2 &\rightarrow bS_1 \mid \varepsilon \end{aligned}$$

Dado que para todas hemos encontrado gramáticas de tipo 3, podemos asegurar que todos los lenguajes generados aquí son de tipo 3.

3. Determinar si el lenguaje sobre el alfabeto A={a,b} generado por la siguiente gramática es regular (justifica la respuesta):

$$S \rightarrow S_1 a S_2 \quad ; \quad S_1 \rightarrow bS_1 \mid \varepsilon \qquad S_2 \rightarrow S_1 \mid baS_2 \mid \varepsilon$$

S_1 depende de sí mismo. Lo analizamos:

$$\mathscr{L}(S_1) = \{b^i \ / \ i \geq 0\}$$

En lenguaje informal diremos que el lenguaje que genera la gramatica S_1 es una sucesión de 0,1,2,... símbolos 'b'.
S_2 depende de S_1. Lo analizamos:

$$\mathscr{L}(S_2) = \{u\epsilon\{a,b\}^* \ / \ (u \ empieza \ por \ 'b' \ \bigwedge N_a(u) \leq N_b(u)) \bigvee u = \varepsilon\}$$

Informalmente, el lenguaje que genera la gramática S_2 es aquél cuyas palabras están compuestas por símbolos 'a' y 'b' de manera que o bien u empieza por 'b' y el número de 'a' es menor o igual que el número de 'b' o bien u es la palabra vacía.

S depende de S_1 y de S_2. Lo analizamos:

$$S \ genera \ las \ siguientes \ cadenas: \begin{cases} bbbbbb....abbbbbb...., & \text{si } S_1 \rightarrow bS_1 \ y \ S_2 \rightarrow S_1 \\ bbbbbb....abab[ab.....], & \text{si } S_1 \rightarrow bS_1 \ y \ S_2 \rightarrow baS_2 \\ bbbbbb....ababa[b.....], & \text{si } S_1 \rightarrow bS_1 \ y \ S_2 \rightarrow S_1 \ y \ S_1 \rightarrow bS_1 \\ bbbbbb....a\varepsilon, & \text{si } S_1 \rightarrow bS_1 \ y \ S_2 \rightarrow \varepsilon \\ \varepsilon abbbbbb...., & \text{si } S_1 \rightarrow \varepsilon \ y \ S_2 \rightarrow S_1 \\ \varepsilon ababa[ba....], & \text{si } S_1 \rightarrow \varepsilon \ y \ S_2 \rightarrow baS_2 \\ \varepsilon ababa[b.....], & \text{si } S_1 \rightarrow \varepsilon \ y \ S_2 \rightarrow S_1 \ y \ S_1 \ \rightarrow \ bS_1 \\ \varepsilon a\varepsilon, & \text{si } S_1 \rightarrow \varepsilon \ y \ S_2 \rightarrow \varepsilon \end{cases}$$

Por tanto podemos decir que:

$$\mathscr{L}(S) = \{u\epsilon\{a,b\}^* \ / \ N_a(u) = N_b(u) + 1 \mid N_a(u) \leq N_b(u) + 1\}$$

Informalmente, el lenguaje generado por la gramática S está formado por palabras que cumplen la condición de que o bien el numero de 'a' es menor que el número de 'b', sin restricciones o bien hay sólo un símbolo 'a' más que símbolos 'b'

Sabiendo ésto y utilizando la metodología de trabajo del ejercicio 2, podemos generar la siguiente gramática de tipo 3 que genera el mismo lenguaje:

$$\begin{aligned} S &\rightarrow aS_1 \mid bS \mid bS1 \\ S_1 &\rightarrow bS_1 \mid \varepsilon \end{aligned}$$

Dado que hemos encontrado una gramática de tipo 3 que genera el mismo lenguaje, podemos asegurar que el lenguaje formado por la gramática de tipo 2 anterior era un lenguaje de tipo 3 (o regular) ya que existe una gramática de tipo 3 que también es capaz de generarlo.