Modelos de Computación
grado en ingeniería informática

Memoria de prácticas

Autor
Vladislav Nikolov Vasilev

Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación

Curso 2018-2019

Contents

1 Prácticas

1.1 Práctica 1

1.1.1 Ejercicio 1

Enunciado. Calcula una gramática libre de contexto que genere el lenguaje $L = \{a^n b^m c^m d^{2n}$ tal que $n, m \geq 0\}$.

Solución

Se define la gramática como una cuádrupla con la forma $G = (V, T, P, S)$, siendo V el conjunto de variables, T el conjunto de elementos terminales, P las reglas de producción y S el símbolo inicial. Se puede definir cada uno de los conjuntos de la siguiente forma:

$$V = \{S, X, Y\}$$

$$T = \{a, b, c, d\}$$

$$P = \{S \to aXdd \mid bYc \mid \varepsilon,\ X \to aXdd \mid bYc \mid add \mid \varepsilon,\ Y \to bYc \mid bc \mid \varepsilon\}$$

$$S = \{S\}$$

Ésta es una gramática de **tipo 2**, ya que a la izquierda aparce una variable sola, sin símbolos terminales, y a la derecha aparece la variable con símbolos terminales tanto por la derecha como por la izquierda, impidiendo por tanto que sea regular por la izquierda o por la derecha.

Con ésta gramática, primero se escoge si se van a empezar a generar una a con la secuencia dd al final, o si directamente se comenzará a generar la secuencia b seguida de c. Si se escoge la primera opción, se ponen tantas a al principio y dd al final como sea necesario, y después se puede escoger si se sigue con las b y c, o si se termina sin poner ninguno de los símbolos anteriores. Si se decide comenzar a poner b y c desde el principio o después de poner todas las a y dd que se quieran, se ponen todas las b y c que se quieran, hasta que se decida terminar la secuencia.

Gracias a las reglas de producción se pueden satisfacer todas las restricciones del lenguaje, ya que por cada a se genera dd, y por cada b se genera c. Además, se puede aceptar la cadena vacía o que alguna de las partes no esté.

1.1.2 Ejercicio 2

Enunciado. Describir una gramática que genere los números decimales escritos con el formato [signo][cifra][punto][cifra]. Por ejemplo, +3.45433, -453.23344, ...

Solución

La solución más sencilla que se puede ofrecer a este problema consiste en utilizar una gramática libre de contexto, como se mostrará a continuación. No obstante, el problema también es resoluble mediante una gramática regular, aumentanto sin embargo el número de producciones y de variables necesarias.

Definimos la gramática como una cuádrupla con la forma $G = (V, T, P, S)$, siendo V el conjunto de variables, T el conjunto de elementos terminales, P las reglas de producción y S el símbolo inicial. Se puede definir cada uno de los conjuntos de la siguiente forma:

$$V = \{S, X\}$$

$$T = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, ., +, -\}$$

$$P = \left\{ \begin{array}{c} S \to +X.X \mid -X.X \\ X \to 0X \mid 1X \mid 2X \mid 3X \mid 4X \mid 5X \mid 6X \mid 7X \mid 8X \mid 9X \mid \\ 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9 \end{array} \right\}$$

$$S = \{S\}$$

Primero se genera el signo y el punto, pudiendo escoger si el número es positivo o negativo. Después, en la parte entera y decimal se van generando números en el rango $[0, 9]$, pudiendo escoger cuál es el siguiente número o cuando terminar de insertar números.

1.1.3 Ejercicio 3

Enunciado. Calcula una gramática libre de contexto que genere el lenguaje $L = \{0^i 1^j 2^k \text{ tal que } i \neq j \text{ o } j \neq k\}$.

Solución

Definimos la gramática como una cuádrupla con la forma $G = (V, T, P, S)$, siendo V el conjunto de variables, T el conjunto de elementos terminales, P las reglas de producción y S el símbolo inicial. Se puede definir cada uno de los conjuntos de la siguiente forma:

$$V = \{S, X, Y, Z, P, R, A, B, M, K, U, D, N, L, C, Q\}$$

$$T = \{0, 1, 2\}$$

$$P = \left\{ \begin{array}{c} S \to 0X1 \mid 0Y2 \mid 1Z2 \mid 0A1P2 \mid 0R1B2,\ X \to 0X \mid X1 \mid \varepsilon \\ Y \to 0Y \mid Y2 \mid \varepsilon,\ Z \to 1Z \mid Z2 \mid \varepsilon,\ P \to P2 \mid \varepsilon,\ R \to 0R \mid \varepsilon, \\ A \to 0M \mid N1,\ M \to OMU \mid \varepsilon,\ U \to 1 \mid \varepsilon,\ N \to CN1 \mid \varepsilon,\ C \to 0 \mid \varepsilon, \\ B \to 1K \mid L2,\ K \to 1KD \mid \varepsilon,\ D \to 2 \mid \varepsilon,\ L \to QL2 \mid \varepsilon,\ Q \to 1 \mid \varepsilon \end{array} \right\}$$

$$S = \{S\}$$

Ya que hay muchas reglas y puede no llegar a quedar claro para qué es cada una, vamos a ir comentándolas para que no queden dudas sobre el por qué de cada una de ellas.

La primera de ellas, $S \to 0X1$, indica que solo se van a producir los símbolos 0 y 1, dándose por tanto la condición $j \neq k$, ya que no hay ningún símbolo 2. X puede ser sustituido por tantos 0 o 1 como se desee, lo cuál corresponde a la producción $X \to 0X \mid X1 \mid \varepsilon$.

Después tenemos la producción $S \to 0Y2$, la cuál es parecida a la anterior, solo que esta vez se producen solo los símbolos 0 y 2, satisfaciendo por tanto las condiciones $i \neq j$ y $j \neq k$ simultáneamente. La variable Y puede ser sustituida por tantos 0 o 2 como se desee, lo cuál se corresponde con la producción $Y \to 0Y \mid Y2 \mid \varepsilon$.

La regla $S \to 1Z2$ permite producir los símbolos 1 y 2. En este caso, esta regla permite satisfacer la restricción $i \neq j$, ya que no se produce ningún símbolo 0. La variable Z puede ser sustituida por tantos 1 y 2 como se desee. Ésto se corresponde con la producción $Z \to 1Z \mid Z2 \mid \varepsilon$.

1.1.4 Ejercicio 4

Enunciado. Una empresa de videojuegos "*The fantastic platform*" están planteando diseñar una gramática capaz de generar niveles de un juego de plataformas, cada uno de los niveles siguiendo las siguientes restricciones:

- Hay 2 grupos de enemigos: grupos grandes (g) y grupos pequeños (p).
- Hay 2 tipos de monstruos: fuertes (f) y débiles (d).
- Los grupos grandes de enemigos tienen, al menos, 1 monstruo fuerte y 1 débil. Y los 2 primeros monstruos pueden ir en cualquier orden. A partir del tercer monstruo, irán primero los débiles y después los fuertes.
- Los grupos pequeños tienen como mucho 1 monstruo fuerte.

- Al final de cada nivel habrá una sala de recompensas (x).

Por ejemplo, la cadena terminal "*gfddddfffpdddfx*" representa que el nivel tiene (*gfddddfff*) un grupo grande con un monstruo fuerte, 4 débiles y otros 3 fuertes; después tiene (*pddddf*) un grupo pequeño con 3 débiles y uno fuerte.

Elaborar una gramática que genere estos niveles con sus restricciones. Cada palabra del lenguaje es un solo nivel. ¿A qué tipo de gramática dentro de la jerarquía de Chomsky pertenece la gramática diseñada?

¿Sería posible diseñar una gramática de tipo 3 para dicho problema?

Solución

1.2 Práctica 2

1.3 Práctica 3

2 Ejercicios voluntarios