Descripción del código

Clase Reader

Esta clase únicamente se encarga de leer un archivo .txt donde debe haber una gramática libre de contexto. Una vez que se abre el archivo, se crea una instancia para poder referenciar a las producciones de la gramática (las producciones se almacenan en una lista).

Clase Chomsky

Clase encargada de convertir una gramática libre de contexto a la forma normal de Chomsky. El proceso hace uso de 4 métodos:

- deleteEpsilon():
  Encuentra las producciones que generen épsilon y posteriormente manda otro método para modificar todas las producciones que involucren épsilon para ese símbolo.
- deleteUnits():
  Elimina las producciones unitarias por cada uno de los símbolos en la gramática.
- deleteThree():
  Elimina las producciones que generen más de dos símbolos terminales o no terminales.
- deleteSymbolsTerminals():
  Separa las producciones que generen un símbolo terminal junto con uno generador para finalizar la conversión a FNC.

Para hacer uso de estos métodos primero se debe instanciar un objeto Chomsky haciendo uso del constructor que recibe una lista de producciones.

Clase ChomskyAnalizer (CYK)

Representa un analizador que utiliza el algoritmo CYK para determinar si una cadena pertenece o no a una FNC. Además, se encarga de solicitar el input necesario y al finalizar imprime el árbol de derivación que se utilizo para llegar al resultado. Consta de 2 métodos principales:

- CYK(Chomksy glc, String input):
  Analiza la cadena ingresada en conjunto con la gramática para verificar si pertenece a esta. Después de inicializar la matriz, se comienza a llenar la diagonal y posteriormente se llena el resto de ella. Durante este proceso se manda a llamar el método findProduction, el cual busca en las producciones por una que genere los símbolos deseados.
  Una vez que se termina de llena la matriz, se verifica que el elemento en la esquina superior derecha contenga el símbolo inicial. Si no lo contiene, muestra al usuario que la cadena fue rechazada. Si sí lo contiene, muestra al usuario que la cadena fue aceptada e imprime el árbol de derivación.

- findPath(ArrayList<Production> prods, String[][] matrix):
  Usando backtracking dentro de la matriz generada previamente, encuentra el camino de símbolos que genere la cadena analizada. Este método funciona como la preparación del método findSymbol, el cual, recursivamente, encuentra el camino mientras genera un árbol binario con los símbolos encontrados.

Todos los métodos dentro de esta clase son estáticos, por lo cual no se debe instanciar un objeto de esta clase para analizar una cadena dentro de una gramática.

Clases auxiliares

Para la implementación de este analizador, se hacen uso de otras dos clases Production y TreeNode.

- Production: representa un símbolo junto con sus posibles producciones.
- TreeNode: representa un nodo de un árbol. Esta clase fue tomada de internet para poder imprimir el árbol de derivación dentro de la consola.