

Institución: Universidad de San Carlos de Guatemala

Carrera: Ingeniería en ciencias y sistemas

Curso: Compiladores 1

Nombre: José Daniel Alvarado Fajardo

Carne: 201904061

Fecha: 07/03/2021



Manual de técnico de proyecto #1

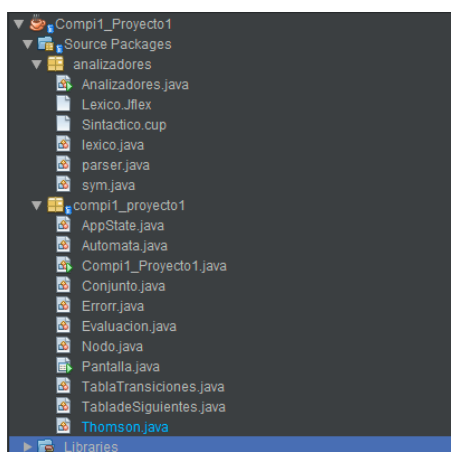
Descripción:

A continuación, se describirán las clases y métodos utilizados para la realización de este proyecto.

Dentro del programa se cuenta con dos paquetes: Analizadores y Compi1_Prroyecto1, el primero de ellos sirve para crear un analizador léxico y otro sintáctico mientras que el segundo sirve para realizar todas las funcionalidades principales del programa

Dentro del paquete de analizadores encontramos lo siguiente:

- Léxico.Jflex: en este archivo es donde se definen los patrones de todos los tokens del lenguaje.
- Sintactico.cup: en este archivo se podrán definir todos los terminales declarados en el archivo de Jflex y los no terminales para poder definir una gramática que nos ayude a reconocer el orden en que deben venir los tokens.



Dentro del paquete de Compi1_Proyecto1 encontramos las siguientes clases:

- **Compi1_proyecto1:** Esta clase es la que inicializa la aplicación mandando a llamar una pantalla en la cual se mostraran todas las opciones a realizar con el programa.
- **AppState:** Dentro de esta clase se definen todas aquellas variables estáticas que nos ayudaran en la ejecución del programa, también se encontraran funciones que nos ayudaran a manejar el estado de esas variables.
 1. **Graficar arboles:** esta función permite graficar todos los árboles generados al momento de realizar el análisis sintáctico.
 2. **GraficarThosmon:** esta función borra todas las imágenes que se encuentren dentro del programa para luego mandar a llamar la clase de Thomson para que pueda graficar los datos.
 3. **Graficar tabla de siguientes:** Este programa se encarga de recorrer todos los árboles creados para poder graficarlos.
 4. **Graficar tablas de transiciones:** Esta función se encarga de recorrer las tablas de siguientes que se encuentren dentro del programa para poder mandar a llamar el método que grafica todas las tablas de transiciones.
 5. **Crear autómatas:** se encarga de mandar a llamar la función que crea los autómatas a partir de una tabla de transiciones.
 6. **Vaciar carpetas:** Esta es la función que se encarga de limpiar todas las imágenes que estén dentro de las carpetas determinadas para los reportes para que el usuario únicamente vea los archivos de las expresiones regulares que se evalúa en ese momento.
- **Autómata:** esta es la clase encargada a todas las funcionalidades referentes a los autómatas. Dentro de esta clase se cuenta con las siguientes funciones:
 1. **Autómata:** este es el constructor de la clase la cual se encarga de instanciar un autómata recibiendo la tabla de transiciones.

2. **Graficar:** esta función se encarga de poder graficar el autómata.
 3. **Validar cadena:** función que se encarga de recibir una cadena y retornar un valor de verdadero en caso e que la cadena fuera valida por el autómata y en caso contrario enviara un valor de falso.
- **Conjunto:** Esta Clase se creado con el objetivo de facilitar el manejo de los mismos teniendo como atributos el nombre del conjunto y sus elementos. Para formar sus elementos cuenta con un constructor el cual se encarga de reconocer los elementos que estar dentro de los valores ascii proporcionados.
 - **Error:** Esta clase se creo con el fin de facilitar el acceso a los atributos de un error, siendo estos: tipo de error, descripción, numero de fila y numero de columna.
 - **Evaluación:** De igualmente se creo la clase de evaluaciones la cual posee su lexema y el nombre de la expresión regular con la que se compara esta.
 - **Nodo:** Esta clase se encarga de recibir los datos proporcionados durante el tiempo en que se ejecuta el análisis sitnatico.
 - **Pantalla:** En esta clase se encarga de renderizar toda la interfaz gráfica, adicionalmente a ello se le ha agregado un método el cual se encarga de desencadenar todas las funcionalidades mencionadas anteriormente.
 - **Tabla de transiciones:** Clase que se encarga de manejar todo lo referente las tablad de transiciones. Cuenta con los siguientes métodos:
 1. **Crear tabla de transiciones:** esta se encarga de poder instanciar las tablas a través de una tabla de siguientes.
 2. **Graficar tabal de siguientes:** esta función se encarga de generar el archivo .dot para luego crear la imagen que corresponde a la tabla.
 - **Tabal de siguientes:** esta clase se encarga de manejar todo lo referente a las tablas de transiciones y desencadenar todas las operaciones referentes a ellas. Dentro de sus funciones se encuentran las siguientes:
 1. **Crear tabla de siguientes:** esta función es la encarga de poder instanciar las tablas a través de los arboles que se encuentren dentro del programa.

- **Thomson:** Esta clase se encarga de manejar todo lo referente a los autómatas de Thomson. Dentro de sus funciones se encuentran las siguientes:
 2. **Código interno:** esa función se encarga de mandar a llamar recursivamente las operaciones de concatenación, viene una vez o mas veces, viene cero o mas veces y viene o no viene.
- Para la realización del analizador léxico se hizo uso de jflex:



- Para la realización del analizador sintáctico se hizo uso de jcup:



Enlaces de descarga:

- <https://jflex.de/>
- <http://www2.cs.tum.edu/projects/cup/>