

MANUAL TÉCNICO

INDICE

Contents

PRESENTACION.....	3
OBJETIVOS	4
PROCESOS	5
DESCRIPCION DE LOS METODOS CREADOS.....	6
Nodo_arbol.java.....	6
Nodo_error.java	6
Nodo_lista_transicion.java.....	6
Reportes.java.....	7
Lexico.java.....	7
Nodo_tabla.java	7
Arboles.java	8
Proyecto.java.....	8

PRESENTACION

El siguiente manual describe los pasos necesarios para que cualquier persona que tenga cierta base de programación en java, manejo de archivos pueda hacer uso y entender el funcionamiento del programa, a su vez guiara a los usuarios que harán soporte al sistema, el cual les dará a conocer los requerimientos y la estructura para la construcción del sistema, además contiene descripción de las expresiones regulares del lenguaje desarrollado.

OBJETIVOS

General:

- Brindar información necesaria para el uso y manejo del programa, con el fin de que se pueda hacer soporte y modificaciones en caso de ser necesarias.

Especificos:

- Describir los métodos creados en el programa
- Describir las clases creadas en el programa

PROCESOS

Procesos de entrada:

Ingresar al programa

Seleccionar el archivo .olc

Procesos de salida:

Reporte de arboles

Reporte de afnd

Reporte de afd

Reporte de tabla siguientes

Reporte de tabla transiciones

Reporte de errores

DESCRIPCION DE LOS METODOS CREADOS

Nodo_arbol.java

Este archivo contiene los datos para cada nodo del árbol a crear, contiene las siguientes variables:

```
public String dato;  
public Nodo_arbol left; //izquierda  
public Nodo_arbol right; //derecha  
public boolean hoja = false;  
public int id;  
public String anulabilidad;  
public ArrayList<Integer> primeros;  
public ArrayList<Integer> ultimos;
```

Cada variable tiene su getter y setter para poder obtener o asignar un valor a cada variable. La variable hoja sirve para verificar si un nodo es una hoja.

Nodo_error.java

Este archivo contiene los datos para el reporte de errores, contiene las siguientes variables:

```
String Simbolo_error;  
int linea;  
int columna;
```

Cada variable tiene su getter y setter para obtener o asignar un valor al nodo error.

Nodo_lista_transicion.java

Este archivo contiene los datos para la creación de la tabla de transiciones y el afd, contiene las siguientes variables:

```
String estado_inicio;
Map<String, ArrayList<Integer>> estado_inicial = new HashMap<>();
String simbolo;
Map<String, Boolean> estado_final = new HashMap<>();
String estado_final1;
```

Cada variable tiene su getter y setter para obtener y asignar un valor a cada variable. Este almacenara el estado inicial, estado final y el símbolo con que ira al estado final.

Reportes.java

Este archivo contiene la lógica para generar los reportes par arboles, tabla de siguientes, tabla de transiciones, afd y afnd. Lo que hace es crear una carpeta donde se creara el reporte, luego lo obtenido se agrega en un archivo .dot para luego convertirlo a .png y asi obtener la imagen del reporte.

Lexico.java

Este archivo contiene la lógica para hacer el análisis lexico, este archivo se creo a partir del archivo lexico.jflex, para eso se debe ejecutar el archivo G_lexico.java

Nodo_tabla.java

Este archivo contiene los datos para el reporte de tabla siguientes. Contiene las siguientes variables:

```
ic class Nodo_tabla {

public String dato;
public int nodo;
ArrayList<Integer> siguientes = new ArrayList<Integer>();
```

Cada variable tiene su setter y getter para obtener o conseguir el valor a cada nodo_tabla. El int nodo guarda el numero de nodo, el string dato guarda el símbolo para el estado y el array list siguientes guarda los siguientes del nodo.

Arboles.java

Este archivo contiene la lógica para crear el afd, tabla siguientes, tabla de transiciones y el árbol. Primero crea el reporte para árbol, con el método "Mostrar_arbol()" mandando como parámetro el arobl y un contador. Este método retornara un string con lo necesario para graficar el árbol. Luego crea el reporte de tabla de siguientes, haciendo uso del método "Reporte_tabla()" mandando como parámetro una tabla con los datos necesarios para crearse.

Luego crea el reporte de la tabla de transiciones usando el método "Tabla_transiciones()" mandando como parámetros esta el árbol, la tabla con los datos de siguientes y una lista_transiciones. Por ultimo crea el afd con el método "Graficar_AFD", mandando como parámetros la lista de transiciones.

Proyecto.java

Este archivo contiene la lógica de cada componente de la interfaz grafica, contiene los siguientes métodos:

- **Crear_archivoActionPerformed:** Este método crea el archivo abriendo un cuadro de dialogo para ingresar el nombre del archivo a crear, luego se crea el archivo vacio.
- **Guardar_action_performed:** Este método cumple con la función de guardar lo que este en el text label para el archivo actual.
- **Leer_archivoActionPerformed:** Este método contiene la lógica para leer el archivo obteniendo la información del text área, primero verifica si la lista de errores esta vacia, de ser asi muestra un mensaje de que no hay errores, caso contrario se muestra un mensaje de que hay errores y luego se procede a crear el reporte de errores.
- **Abrir_archivoActionPerformed:** Este método contiene la lógica para abrir un archivo, para eso abre un cuadro de dialogo para obtener el archivo, luego extrae todo su contenido para mostrarlo en un text área.
- **Guardar_comoActionPerformed:** Este método contiene la lógica para guardar como un archivo, para esto se abre un cuadro de dialogo para ingresar el

nombre del nuevo archivo, luego el contenido del text área pasara al archivo creado