

Practica Teoría de lenguajes Autómatas Finitos

Manual Tecnico

Alejandro Isaza Delgado
Santiago Ramirez

Programa que se encarga de manejar autómatas finitos dinámicamente y también de poder guardarlos en un archivo .txt para poder usarlos después y con la posibilidad de poder probarlos, Todas las clases usadas están en el paquete practica.

Package practica

Class Summary

Class	Description
Archivos	Clase que se encarga de manejar la lectura y escritura de los archivos txt
Automata	Clase para representar un automata en la cual se almacenan los estados, los simbolos de entrada, estado inicial del automata y sus estados de aceptacion con sus respectivos setters y getters
FormularioCrearAutomata	Jframe principal en el cual se maneja toda la manipulacion del automata dinamicamente
FormularioInicial	JFrame inicial el cual tienen dos botones uno para crear un nuevo automata o para cargar un automata guardado previamente
TestAutomata	JFrame en el cual dado un automata que se pasa por el constructor lo probamos haciendo uso de sus metodos

Class Archivos

java.lang.Object
practica.Archivos

```
public class Archivos
extends java.lang.Object
```

Clase que se encarga de manejar la lectura y escritura de los archivos txt

Constructor Summary

Constructors

Constructor and Description
Archivos()

Method Summary

All Methods	Instance Methods	Concrete Methods
-------------	------------------	------------------

Modifier and Type	Method and Description
boolean	<code>creartxt(java.lang.String direccion, java.lang.String archivo, java.lang.String Nombre)</code> Metodo que se encarga de crear un archivo txt dada una direccion, un nombre para el archivo y su contenido
java.lang.String	<code>leertxt(java.lang.String direccion, java.lang.String codificacion)</code> metodo que se encarga de leer un archivo txt dada una direccion y una codificacion

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Class Automata

java.lang.Object
practica.Automata

```
public class Automata
extends java.lang.Object
```

Clase para representar un automata en la cual se almacenan los estados, los simbolos de entrada, estado inicial del automata y sus estados de aceptacion con sus respectivos setters y getters

Constructor Summary

Constructors
Constructor and Description
Automata()

Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method and Description	
java.lang.String	getEstadoInicial()	
java.util.HashMap<java.lang.String,java.util.HashMap>	getEstados()	
java.util.ArrayList	getEstadosAceptacion()	
java.util.ArrayList	getSimbolos()	
boolean	isValid(java.lang.String cadena) Metodo el cual se encarga de validar una cadena en el automata y decir si esta si queda en un estado de aceptacion o no	
void	setEstadoInicial(java.lang.String estadoInicial)	
void	setEstados(java.util.HashMap<java.lang.String,java.util.HashMap> estados)	
void	setEstadosAceptacion(java.util.ArrayList estadosAceptacion)	
void	setSimbolos(java.util.ArrayList simbolos)	

Clase FormularioCrearAutomata

Hereda de JFrame

Constructor

```
public FormularioCrearAutomata()
```

Métodos

```
public void transformarAutomata(Automata automata)
```

Método que se encarga de llenar las tablas del formulario con sus respectivos símbolos y estados dado una objeto de la clase Autómata

```
public void removeColumnAndData(JTable table, int vColIndex)
```

Remueve una columna de la tabla y sus datos

```
public Automata crearAutomata()
```

Método que se encarga de dado los estados y los símbolos de las tablas crear un objeto de la clase Autómata y llenarlos respectivamente

```
private void
```

```
AgregarSimboloActionPerformed(java.awt.event.ActionEvent  
evt)
```

Método que agrega un símbolo a la tabla dado una carácter

```
private void
```

```
EliminarSimboloActionPerformed(java.awt.event.ActionEvent  
evt)
```

Método que elimina el símbolo que este seleccionado en la tabla

private void

AgregarEstadoActionPerformed(java.awt.event.ActionEvent evt)

Método que Agrega un estado a la tabla dado una cadena

private void

EliminarEstadoActionPerformed(java.awt.event.ActionEvent evt)

Método que elimina el estado de la tabla que este seleccionado

private void

ProbarAutomataActionPerformed(java.awt.event.ActionEvent evt)

Método que invoca el método crearAutomata() y crear una instancia de otro JFrame mandándole el autómata para probarlo

private void

GuardarAutomataActionPerformed(java.awt.event.ActionEvent evt)

Método que Guarda el autómata usando la librería Gson y la clase de Archivos

Clase FormularioInicial

Constructor

```
public FormularioInicial()
```

Métodos

```
private void cargarActionPerformed(java.awt.event.ActionEvent  
evt)
```

Método que carga un autómata guardado previamente haciendo uso de la librería Gson y la clase Archivos, luego crea una instancia de la clase FormularioCraerAutomata y le manda el autómata

```
private void crearActionPerformed(java.awt.event.ActionEvent  
evt)
```

Crea una instancia de la clase FormularioCrearAutomata

Clase TestAutomata

Constructor

`public TestAutomata(java.awt.Frame parent, boolean modal)`

`public TestAutomata(java.awt.Frame parent, boolean modal, Automata automata)`

Métodos

`private void release(java.awt.event.KeyEvent evt)`

Método el cual se encarga de cada que el usuario digita un caracter mandar la cadena que hay en el Jtextfield para verificarla

`private void type(java.awt.event.KeyEvent evt)`

Método el cual se encarga de cada que el usuario digita una tecla esta esté dentro de los símbolos de entrada del autómata sino no la deja ingresar al Jtextfield

Librerías Usadas

Gson 2.2.2

Gson (también conocido como Google Gson) es una biblioteca de código abierto para el lenguaje de programación Java que permite la serialización y deserialización entre objetos Java y su representación en notación JSON.

La podemos encontrar en este enlace:

<http://www.java2s.com/Code/Jar/g/Downloadgson222jar.htm>