



Universidad de Alcalá

Escuela Politécnica Superior

Universidad de Alcalá

Procesadores del lenguaje

PECL1

Martes 17:00 – 19:00

Grado en Ingeniería Informática – Curso 2019/2020

Marcos Barranquero Fernández – 51129104N

INTRODUCCIÓN

Se ha realizado un programa que, dada una matriz de transición de estados, permite realizar las siguientes operaciones.

1. Verificar si una cadena dada cumple con la expresión regular asociada.
2. Imprimir cadenas que cumplan la expresión regular, pasando como argumento la cantidad de cadenas a generar y la longitud máxima de estas.
3. Imprimir la matriz.

EJECUCIÓN SECUENCIAL

1. Se crea un AFD a partir de una expresión regular en JFLAP. Se guarda el archivo asociado.
2. Se crea una instancia de la clase Generador de matrices. Esta clase recibe la ruta del archivo de JFLAP, y lee de este los nodos y las transiciones entre nodos, así como el nodo inicial y nodos finales.
3. Se crea un autómata, que recibe los estados y transiciones del generador y crea la matriz de saltos. Además, carga el estado inicial y los estados finales.
4. Se despliega el menú de opciones mencionado en la introducción.
 - a. Si se elige verificar cadena, se crea una máquina de estados que valida recursivamente la cadena dada, carácter a carácter.
 - b. Si se elige generar cadenas, parte del estado inicial y recursivamente recorre todos los estados posibles. Si se llega a un estado final y no se supera el tamaño máximo de cadena, se imprime por pantalla la cadena y se suma al contador de cadenas.
 - c. Si se elige imprimir la matriz, se recorren los estados de la matriz y para cada estado se imprimen todos los saltos posibles.

EXPLICACIÓN MODULAR

El programa se divide en las siguientes clases:

CLASE TRANSICIÓN

Abstracción de una transición de un nodo origen a uno destino, recibiendo un carácter asociado.

CLASE GENERADOR DE MATRIZ

Recibe la ruta del archivo de JFlap y lee el archivo línea a línea.

- En el método de lectura de estados, si encuentra un estado, lo añade al arraylist de estados.
- En el método de lectura de transiciones, se leen estados origen, destino y caracteres de transición, y va generando transiciones y guardándolas en el array de éstas.
- Tenemos dos getters que toman el estado inicial y los estados finales.

CLASE GENERADOR DE CADENAS

Esta clase contiene el autómata asociado, un contador de cadenas generada y un ArrayList de cadenas generadas. Tenemos un método envoltorio que llama al método recursivo, pasando la longitud máxima y el número de cadenas generado. Mientras el numero de cadenas generado sea menor que el de cadenas solicitadas y haya estados por recorrer, va saltando a nuevos estados y pasando por argumento la cadena acumulada.

Una vez alcanza un estado final que no se haya llegado antes, lo imprime por pantalla.

Finaliza cuando no se pueden generar más cadenas o se alcanza el número de cadenas deseado.

VERIFICADOR DE CADENAS

Esta clase contiene el autómata asociado y el estado actual. Al recibir una cadena, el estado actual se establece a 0, y recorre carácter a carácter validando la cadena. Si llega hasta el final y no ha caído en error, devuelve True.

CLASE AFD – AUTÓMATA FINITO DETERMINISTA

Clase que representa un autómata finito determinista. Contiene la matriz de transiciones de estado, el estado inicial y los estados finales, así como el alfabeto usado por el autómata. También tiene un método que permite verificar si un salto es posible para esa ER asociada, un método para cargar el alfabeto, y otro para ver si el estado pasado es el estado final.