



Práctica Teoría de Autómatas

Alejandro Moreno Núñez de Arenas
Javier Oliver Díaz-Alejo
María Dolores Sesmero Pozo

1. Explicación de la práctica.

1.1 Extracción de la tabla.

Para llevar a cabo la extracción de la tabla se genera un archivo .fle, el cual, teniendo previamente el código fuente de la página de la bolsa de Madrid, recorrerá dicho código e irá extrayendo la tabla deseada.

Para realizar esta tarea, en primer lugar se hace una definición de lectura de archivos en la parte del fichero de flex correspondiente a las Declaraciones.

A continuación, se definen las dos reglas que nos van a permitir leer la tabla mencionada anteriormente.

- La primera, se encarga de reconocer el identificador de la tabla “*ctl00_Contenido_tblAcciones*”. Cuando reconoce dicha cadena, se empieza a volcar la información en una cadena auxiliar.
- Cuando se encuentra la segunda regla definida (“*</table>*”), correspondiente al fin de la tabla, se corta el volcado a la cadena.

Una vez se ha completado la copia de la tabla a la cadena, se procede a volcar toda la información leída a un fichero de salida (“*Tabla.html*”).

1.2 Definición del léxico.

Para la definición del léxico usamos una serie de estados, dentro de los cuales, se definen una serie de reglas que usaremos para reconocer los elementos de la tabla.

- YYINITIAL: Se reconoce el inicio y final del fichero *Tabla.html*.
- Tabla: Reconoce el inicio de la tabla guardada en el archivo y que contiene los campos de los cuales queremos obtener la información.
- Fila: Analiza cada una de las filas. Las filas pueden ser de dos tipos:
 - Filas tipo cabecera: Contiene los nombres de cada uno de los campos. Se analiza a través del estado CABECERA, recorriéndola hasta donde acaba la fila.
 - Filas tipo datos: Va recorriendo los distintos estados de reconocimiento y extrayendo los datos, para que más tarde puedan ser usados por el fichero .cup:
 - Nombre: Reconoce el nombre de la empresa.
 - Ult: Reconoce el campo Último de la tabla.
 - Dif: Reconoce el campo %Dif.
 - Max: Reconoce el campo Máx.
 - Min: Reconoce el campo Min.
 - Vol: Reconoce el campo Volumen.
 - Efec: Reconoce el campo Efectivo
 - Fecha: Reconoce el campo Fecha.
 - Hora: Reconoce el campo Hora.

Se debe añadir que al reconocer el final de un campo (Cadena “*</td>*”) se realiza la transición al siguiente estado, siguiendo el orden indicado anteriormente.

1.3 Definición de la gramática.

Para la definición de la gramática usamos la herramienta Cup, a través de la cual realizaremos el análisis semántico.

En primer lugar, definimos lo que será el main cuando se transforme el código a Java. En dicho main se invoca al fichero .fle que se encarga de la definición del léxico.

A continuación, a través del archivo flex se va reconociendo los elementos léxicos y se van definiendo como terminales (aquellos que pueden terminar una regla semántica) y los no terminales.

Una vez definidos tanto terminales como no terminales, se pasa a la creación y definición de las reglas semántica, que establecen la composición de cada uno de los elementos de nuestro analizador.

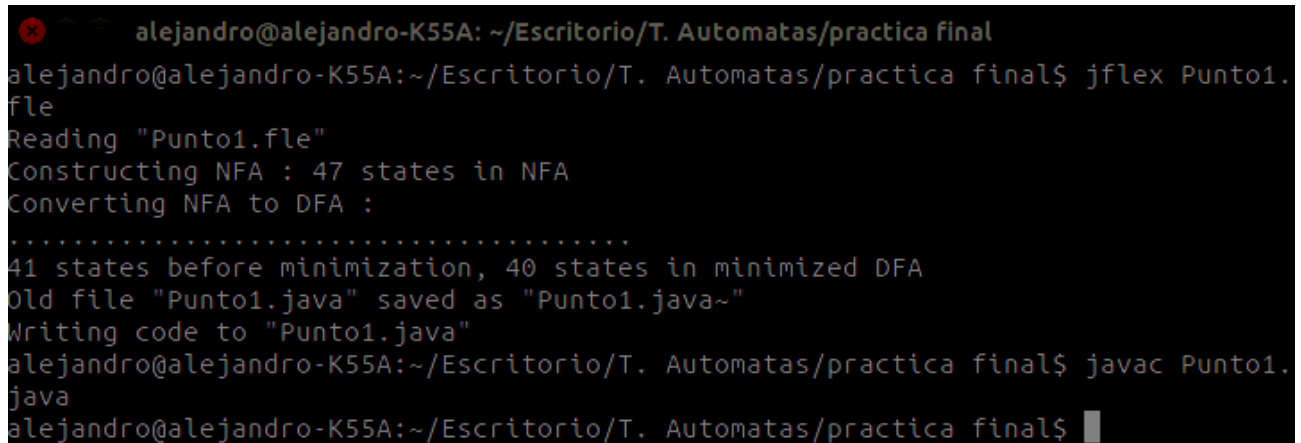
2. MANUAL DE USUARIO

En primer lugar se debe ejecutar el fichero “*Punto1.fle*”. Para ello abriremos un terminal y nos situaremos en la carpeta en el que se encuentre dicho archivo mediante el comando *cd*.

A continuación, compilamos el archivo mediante los comandos:

- `jflex Punto1.fle`
- `javac Punto1.java`

Debiendo obtener unos resultados parecidos a los siguientes:

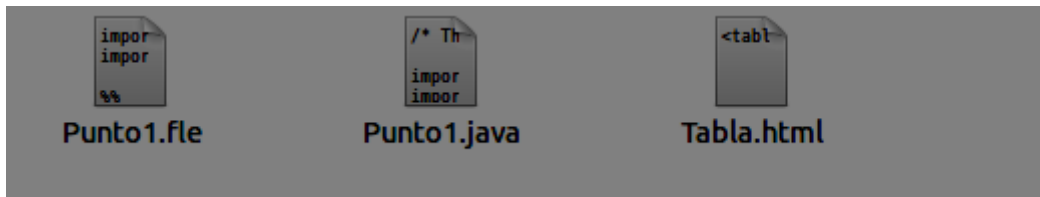


```
alejandro@alejandro-K55A: ~/Escritorio/T. Automatas/practica final
alejandro@alejandro-K55A:~/Escritorio/T. Automatas/practica final$ jflex Punto1.fle
Reading "Punto1.fle"
Constructing NFA : 47 states in NFA
Converting NFA to DFA :
.....
41 states before minimization, 40 states in minimized DFA
Old file "Punto1.java" saved as "Punto1.java~"
Writing code to "Punto1.java"
alejandro@alejandro-K55A:~/Escritorio/T. Automatas/practica final$ javac Punto1.java
alejandro@alejandro-K55A:~/Escritorio/T. Automatas/practica final$
```

Una vez compilado realizamos la ejecución. Para ello debemos de tener guardado un archivo con extensión html que contenga el código fuente de la página de la bolsa de Madrid.

Así, mediante el comando “`java Punto1 nombre_archivo_codigoFuente`”, obtendremos un nuevo archivo html llamado *Tabla.html*, con el cual trabajaremos más tarde.

```
alejandro@alejandro-K55A:~/Escritorio/T. Automatas/practica final$ java Punto1 c
odigoFuente.html
```



Una vez completado el primer paso, pasaremos a la compilación y ejecución para el reconocedor semántico.

Para ello, en primer lugar debemos compilar el reconocer léxico y el semántico.

En el mismo terminal que teníamos abierto, ejecutaremos el comando: *“jflex punto2.fle”*.

Seguiremos ejecutando el comando: *“cup practica2.cup”*, y tras de esto: *“javac Yylex.java sym.java parser.java”*.

De ésta forma obtendremos la compilación del programa, quedándo listo para la ejecución.

```
alejandro@alejandro-K55A:~/Escritorio/T. Automatas/practica final$ jflex punto2.
fle
Reading "punto2.fle"
Constructing NFA : 609 states in NFA
Converting NFA to DFA :
.....
.....
225 states before minimization, 166 states in minimized DFA
Writing code to "Yylex.java"
alejandro@alejandro-K55A:~/Escritorio/T. Automatas/practica final$ cup practica2
.cup
----- CUP v0.11a beta 20060608 Parser Generation Summary -----
 0 errors and 0 warnings
20 terminals, 5 non-terminals, and 15 productions declared,
producing 45 unique parse states.
 0 terminals declared but not used.
 0 non-terminals declared but not used.
 0 productions never reduced.
 0 conflicts detected (0 expected).
Code written to "parser.java", and "sym.java".
----- (v0.11a beta 20060608)
alejandro@alejandro-K55A:~/Escritorio/T. Automatas/practica final$ javac Yylex.j
ava sym.java parser.java
```

Tras la compilación correcta podremos proceder a la ejecución. Sin cambiar de terminal ejecutamos el comando *“tabla parser Tabla.html”*, obteniendo por pantalla los datos de aquellas empresas cuya diferencia es negativa.

```
alejandro@alejandro-K55A:~/Escritorio/T. Automatas/practica final$ java parser Tabla.html
Empresa ABERTIS valor ultimo 14,7250 diferencia -0,41
Empresa ACCIONA valor ultimo 69,9400 diferencia -0,36
Empresa ACERINOX valor ultimo 10,3150 diferencia -0,48
Empresa ACS valor ultimo 28,8750 diferencia -1,84
Empresa AENA valor ultimo 124,5500 diferencia -0,56
Empresa AMADEUS valor ultimo 39,7400 diferencia -0,10
Empresa ARCELORMIT valor ultimo 4,9470 diferencia -3,08
Empresa BA.POPULAR valor ultimo 2,3750 diferencia -4,77
Empresa BA.SABADELL valor ultimo 1,6690 diferencia -2,23
Empresa BA.SANTANDER valor ultimo 4,4210 diferencia -4,45
Empresa BANKIA valor ultimo 0,8130 diferencia -3,56
Empresa BANKINTER valor ultimo 6,6560 diferencia -1,44
Empresa BBVA valor ultimo 5,9830 diferencia -4,77
Empresa CAIXABANK valor ultimo 2,6290 diferencia -3,56
```