



---

# *Procesadores del Lenguaje*

Curso 2019-2020

*Introducción al entorno, análisis léxico y sintáctico*

---

## 1 Introducción

---

Esta primera sesión de laboratorio tiene como propósito lograr que el alumno se familiarice con el entorno de desarrollo elegido y además comenzar con la aplicación de sus conocimientos para el diseño de expresiones regulares y gramáticas básicas con un ejemplo sencillo.

En primer lugar, se analizará un pequeño compilador constituido por un analizador léxico y uno sintáctico capaz de leer la entrada con expresiones numéricas y calcular el resultado de las operaciones. Para ello, crearemos un proyecto Java usando el framework de desarrollo Eclipse<sup>1</sup> –debidamente equipado con la herramienta JFlex y CUP para generar analizadores léxicos y sintácticos.

## 2 Herramientas JFlex/CUP + eclipse

---

Como hemos mencionado en el curso, los trabajos prácticos de esta asignatura se desarrollarán en el lenguaje Java, y van a basarse en las herramientas JFlex (análisis léxico) y CUP (análisis sintáctico).

La primera de ellas, JFlex<sup>2</sup>, es un generador de analizadores léxicos (o *scanners*) para lenguaje Java. En muchos aspectos es similar a su homóloga Flex (Fast LEXical analyzer generator), con la excepción de que esta última ha sido desarrollada en y para lenguaje C. Flex a su vez es una alternativa libre a Lex<sup>3</sup>.

CUP, por otro lado, sirve para generar analizadores sintácticos (o *parsers*) a partir de la especificación de una gramática independiente del contexto. De igual manera que sucede con JFlex, CUP es la alternativa Java a las herramientas YACC<sup>4</sup> y Bison<sup>5</sup> para lenguaje C.

Se recomienda en lo sucesivo tener los manuales de las dos herramientas, JFlex<sup>6</sup> y CUP<sup>7</sup>.

La mayor ventaja que aporta el uso de estas piezas de software es la comodidad de implementación: únicamente se necesitan dos archivos, que agrupan la especificación de *scanner* y *parser* usando una sintaxis clara y simple.

Originalmente, tanto JFlex como CUP se proporcionan como archivos ejecutables. Para generar un scanner con JFlex, el programador escribe un archivo de especificación que mezcla fragmentos en código Java con otras instrucciones propias de la herramienta JFlex. El ejecutable de JFlex procesa el archivo de especificación y genera uno o varios ficheros en código

---

<sup>1</sup> <http://www.eclipse.org/downloads/>

<sup>2</sup> <http://jflex.de/>

<sup>3</sup> [http://en.wikipedia.org/wiki/Lex\\_programming\\_tool](http://en.wikipedia.org/wiki/Lex_programming_tool)

<sup>4</sup> <http://en.wikipedia.org/wiki/Yacc>

<sup>5</sup> [http://en.wikipedia.org/wiki/GNU\\_bison](http://en.wikipedia.org/wiki/GNU_bison)

<sup>6</sup> <http://jflex.de/manual.html>

<sup>7</sup> <http://www.cs.princeton.edu/~appel/modern/java/CUP/manual.html>

Java. Posteriormente será necesario compilar estos archivos Java para producir el bytecode equivalente.

Sin embargo, para una mayor comodidad del programador, existe un plug-in para el entorno de desarrollo Eclipse llamado CUP Eclipse Plugin<sup>8</sup>. Este plug-in nos proporciona plantillas para los archivos JFlex/CUP y simplifica la generación del proyecto realizando la compilación de forma casi automática.

Este plugin requiere Java 1.8 o posterior y Eclipse 4.4 o posterior. El acabado de este plugin aún no es definitivo, por lo que no dispondremos de coloreado de sintaxis para los ficheros JFlex ( sí lo proporciona para CUP), y tendremos que hacer algunas adaptaciones para tabajar únicamente con JFlex como veremos más adelante.

### 3 Creación de un proyecto JFlex-CUP

---

A continuación, indicamos los pasos a seguir para crear un analizador léxico y sintáctico con las herramientas utilizadas.

#### 3.1 Configurando el entorno

---

Para configurar el entorno de desarrollo será necesario llevar a cabo los siguientes pasos:

- Descargar e instalar Java Development Kit (versión 1.8 o posterior, y es **importante instalar el JDK**, ya que JRE no incluye el compilador)
- Descargar e instalar Eclipse (versión 4.4 o superior)
- Instalar el plugin CUP los pasos son:
  - Help -> Install new software...
  - Añadir una nueva localización: Add...-> pegar la URL del sitio de actualizaciones del plug-in CUP<sup>9</sup> y ponerle un nombre ->Aceptar
  - En la ventana anterior, seleccionar el repositorio recién creado y checkear el plug-in CUP. Aceptar y seguir el proceso de instalación.

#### 3.2 Creación del proyecto

---

Una vez instalado, debemos crear un nuevo proyecto JavaCUP de la forma usual en Eclipse: File -> New -> Other... y seleccionando el wizard "CUP Java Project"

Pulsamos "Next>", le damos un nombre a nuestro proyecto y nos aseguramos de que utiliza la JDK 1.8

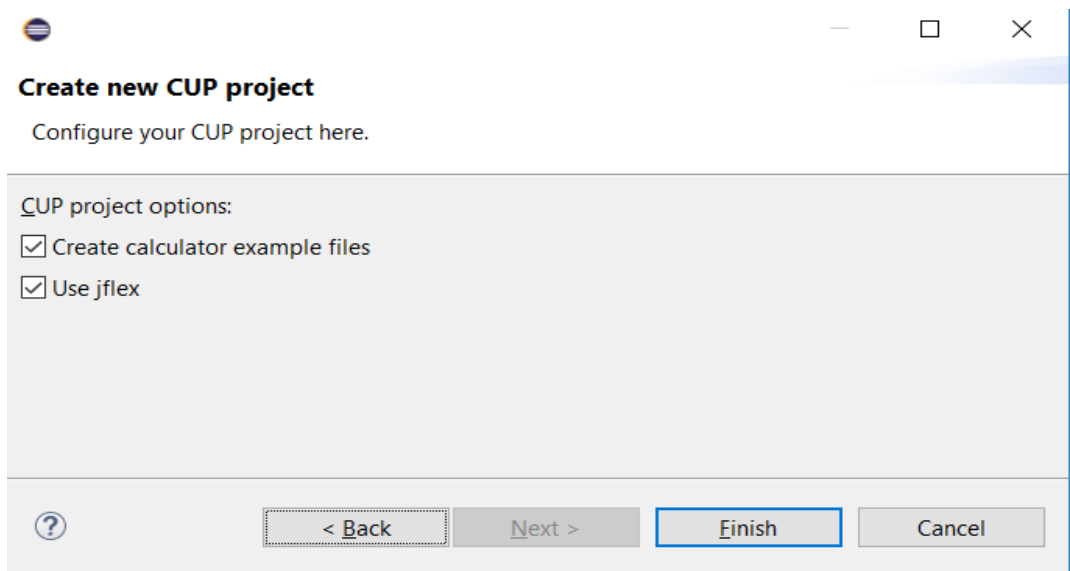
---

<sup>8</sup> <http://www2.cs.tum.edu/projects/cup/eclipse.php>

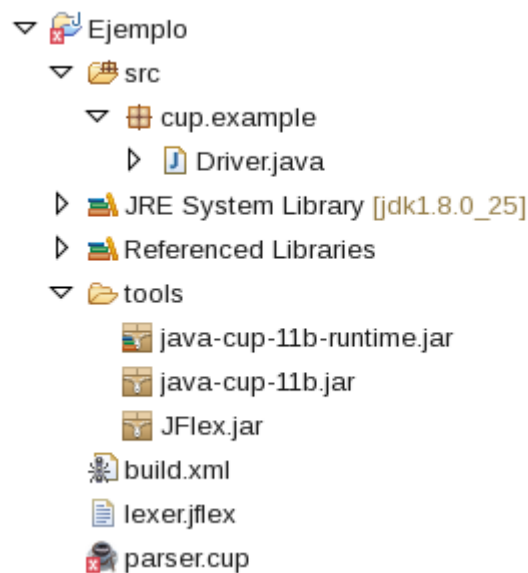
<sup>9</sup> <http://www2.in.tum.de/projects/cup/eclipse>



Pulsamos "Next>" dos veces hasta que nos da la opción de seleccionar que queremos utilizar jFlex, y además vamos a crear el ejemplo de calculadora:



Pulsamos "Finish" y se nos creará el proyecto:



En este proyecto podemos ver tres partes fundamentales:

- Una clase java denominada "Driver" que nos permitirá ejecutar nuestros proyectos.
- Una carpeta tools que contiene los ejecutables de JFlex y CUP. Estos ejecutables son los que utiliza el script de ant denominado "build.xml" para generar el scanner y el parser, y compilarlo todo.
- Los ficheros jflex y cup con la definición de los analizadores léxico y sintáctico, respectivamente.

Por tanto, tenemos un primer proyecto JFlex-CUP con un sencillo reconocedor de expresiones matemáticas que opera con los números y calcula el resultado. El lenguaje objetivo es la definición de operaciones matemáticas simples, siendo el tipo de datos con el que se trabaja es el de números enteros sobre los cuales se ejecutarán operaciones de suma, resta, multiplicación y división, con posibilidad de agrupar con paréntesis. Ejemplo:

```
Ⓢ 4 * 5 + (10 + 10) * 2;
```

```
Ⓢ 40
```

(el programa muestra por pantalla el resultado)

A continuación, en la primera práctica se solicitará modificar la especificación de los lenguajes utilizados en el escáner (JFlex) y parser (CUP) para realizar las siguientes tareas. Como se indicó antes es fundamental tener a la vista los manuales de estas herramientas.