Práctica. Primera Fase

Desarrollo manual de un analizador léxico para un lenguaje sencillo

Considera un lenguaje de programación sencillo en el que los programas están formados por: (i) una sección de *declaraciones*, y (ii) una sección de *instrucciones*. Ambas secciones están separadas por *&&*. La sección de declaraciones, por su parte, está compuesta por una o más declaraciones, separadas por *puntos y coma*. Cada declaración consta de: (i) un *nombre de tipo*, (ii) un *nombre de variable*. Los nombres de tipo pueden ser **int**, **real** y **bool**. Por su parte, los nombres de variable comienzan necesariamente por una letra, seguida de una secuencia de cero o más letras, dígitos, o subrayado (_). Por su parte, la sección de instrucciones consta de una o más instrucciones separadas por *puntos y coma*. El lenguaje únicamente considera instrucciones de *asignación*. Dichas instrucciones constan de una *variable*, seguida de =, seguida de una expresión. Las expresiones básicas consideradas son números enteros con y sin signo, números reales con y sin signo, variables, y **true** y **false**. Los números enteros comienzan, opcionalmente, con un signo + o -. Seguidamente debe aparecer una secuencia de 1 o más dígitos (no se admiten ceros no significativos a la izquierda). Por su parte, los números reales tienen, obligatoriamente, una parte entera, cuya estructura es como la de los números enteros, seguida de bien una parte decimal, bien una parte exponencial, o bien una parte decimal seguida de una parte exponencial. La parte decimal comienza con un ., seguido de una secuencia de 1 o más dígitos (no se permite la aparición de ceros no significativos a la derecha). Por último, y también opcionalmente, puede aparecer una parte exponencial (*e* o *E*, seguida, opcionalmente, de un exponente, cuya estructura es igual que la de los números enteros). Los operadores que pueden utilizarse en las expresiones son los operadores aritméticos binarios usuales (+,-,* y /), el *menos* unario (-), los operadores lógicos **and**, **or** y **not** y los operadores relacionales (<,>,<=,>=,==,!=). También es posible utilizar paréntesis para alterar las precedencias y asociatividades de los operadores.

Ejemplo de programa

```
real peso;
bool pesado
&&
peso = (45.0 * 12e-56) / -002.00;
pesado = (peso >  10.00) or (peso / 002 <= +04)
```

Trabajo a realizar:

Desarrollo de un analizador léxico para el lenguaje descrito. Para ello, deberá entregarse:

- Una memoria con las siguientes secciones:
  - Portada en la que aparezcan los nombres y apellidos de los integrantes del grupo, y el número de grupo.
  - Enumeración de las clases léxicas del lenguaje. Para cada clase debe incluirse, además, una descripción informal, en lenguaje natural.
  - Una especificación formal del léxico del lenguaje mediante definiciones regulares.
  - Diseño de un analizador léxico para el lenguaje mediante un diagrama de transiciones
- Una implementación manual, en Java, del analizador léxico. Debe proporcionarse, además, un programa de prueba que acepte como argumento el archivo a procesar, y genere como salida una descripción legible de la secuenciad de tokens reconocida, y, si procede, un mensaje legible de error léxico detectado.
- Otra implementación del analizador léxico utilizando, para ello, la herramienta JLex. Esta implementación debe proporcionar también un programa de prueba análogo al de la implementación manual.

Fecha límite de entrega: **Domingo 23 de febrero de 2019, a las 11:59 pm**.

Modo de entrega: A través del campus virtual, en un único .zip, que debe tener por nombre el número de grupo (por ejemplo, 01.zip, si el número de grupo es el 01). Dicho archivo de contener: (i) un documento PDF `memoria.pdf` con la memoria; (ii) una carpeta `implementación_manual`, en el interior de la cuál debe incluirse la implementación manual del analizador léxico; (iii) una carpeta `implementación_jlex`, en el interior de la cuál debe incluirse la implementación jlex del analizador léxico; (iv) una carpeta `pruebas` con distintos programas de prueba que permiten probar el analizador léxico. La entrega debe ser realizada solamente por un miembro del grupo.