

## UTN - FRBA

### Sintaxis y Semántica de los Lenguajes

#### Trabajo Práctico n° 2 - FLEX para reconocimiento de categorías léxicas de C

**Implementar en C** un programa que permita reconocer en un *archivo.c* de entrada **todas las categorías léxicas de C, comentarios**. El mismo deberá dar como salida en pantalla un *reporte*. Para desarrollar dicho programa deberá utilizar LEX/FLEX para la generación del código C que implemente un analizador léxico que cumpla con los requisitos.

*Categorías léxicas de C a reconocer:*

- ❖ Constantes (Entera decimal, entera octal, entera hexadecimal, real, caracter)
- ❖ Literal cadena
- ❖ Palabras reservadas (distinguir entre tipo de dato, estructuras de control y otros)
- ❖ Identificadores
- ❖ Caracteres de puntuación/operadores de C.

*Comentarios (en realidad son reconocidos por el PREPROCESADOR pero lo incorporamos)*

- ❖ Comentarios de una línea
- ❖ Comentarios de múltiples líneas

El reporte deberá contener lo siguiente:

*Reporte*

- ❖ Listado de identificadores encontrados indicando la cantidad de veces que aparece cada uno de ellos. El listado debe estar ordenado alfabéticamente.
- ❖ Listado de literales cadena encontrados indicando la longitud de los mismos (sin incluir comillas) y ordenados por longitud ascendente. En caso de igual longitud no importa el orden. Si un literal cadena aparece más de una vez, no volverlo a agregar a la lista.
- ❖ Listado de palabras reservadas en el orden en el que han aparecido en el archivo. Si una palabra reservada aparece de forma repetida, se debe agregar a la lista por cada vez que aparece.
- ❖ Listado de constantes indicando según su tipo:
  - Para las constantes octales indicar su valor entero decimal.
  - Para las constante hexadecimales indicar su valor entero decimal.
  - Para las constantes decimales indicar el valor de cada una y el total acumulado de sumar todas ellas.
  - Para las constantes reales indicar el valor de su mantisa y parte entera.

- Para las constantes carácter, enumerarlas según orden de aparición.
- ❖ Listado de operadores/caracteres de puntuación indicando cantidad de veces que aparecen.
- ❖ Listado de comentarios encontrados distinguiendo si se trata de comentarios de una línea o múltiples líneas.
- ❖ Listado de cadenas y/o caracteres no reconocidos indicando el número de línea donde se encontraron. Se recomienda empezar detectando caracteres no reconocidos individualmente y luego adicionar lógica para formar una cadena con los caracteres consecutivos (cadena).

La entrega de este trabajo práctico es obligatoria, su **fecha límite para consulta, entrega y revisión es el día Domingo 20 de Agosto**. Luego de esa fecha, no se aceptarán más trabajos.

El **entorno de programación queda a criterio de cada grupo de trabajo** (Eclipse, Dev, Codeblocks, **Visual Studio Code** ). Se recomienda un IDE que esté integrado con Git para poder realizar el trabajo en equipo de una forma más práctica.

Formará parte de la entrega, como mínimo, el **archivo del analizador léxico (.l)**, **archivo de funciones desarrolladas (.h o .c)**, **archivo fuente del analizador léxico generado por lex-flex (.c)** y el **archivo ejecutable (.exe)**. Todo debe estar en el repositorio GitHub.

La **entrega será a través del repositorio** de GitHub en la carpeta correspondiente a cada TP.

Las **consultas** podrán ser respondidas a través Discord. Es importante que los utilicen para compartir sus dudas con el resto de los equipos.