



Trabajo Práctico N° 1

Introducción a un scanner

En este TP vamos a practicar manejo de caracteres y escribir en varios fuentes. El objetivo es leer un texto de entrada y separar en tokens. Vamos a considerar 3 posibles tokens

- FDT (por fin de texto) para cuando se termina el archivo (o sea, se leyó EOF).
- SEP (por separador) solo la coma (', ') pertenece a esta categoría.
- CAD (por cadena) es cualquier sucesión de caracteres que no sea ninguno de los tokens anteriores y no pertenezca al conjunto de espacios (espacio, tabulador, etc).

El conjunto de espacios será ignorado.

Si bien no lo hemos explicado en detalle cada token se codifica con un entero, use enum para codificar los tokens.

Vamos a dividir en 2 fuentes, main.c y scanner.c (además será necesario crear scanner.h). En main.c solo vamos a tener la función main que llama a la función `get_token` (que estará en scanner.c) y muestra en pantalla los resultados devueltos: que token es y el lexema correspondiente.

En el repositorio solo suban los archivos fuente (main.c, scanner.c y scanner.h)

Comentarios:

- Lea el texto usando `getchar`, no utilice ninguna otra función para ingresar el texto
- Recuerde que en `ctype.h` tiene la función `isspace`
- También le resultará útil usar la función `ungetc`
- Debe compilar adhiriendo estrictamente al estándar (C11 o C17) y no debe usar funciones propias de un sistema operativo en particular.

El archivo `entrada.txt` tiene un texto para hacer pruebas, lo puede copiar y pegar. O puede redireccionar `stdin` a ese archivo. Por ejemplo ejecutando desde la consola algo como (suponiendo que el ejecutable se llama `tp0`):

```
$ ./tp0 <entrada.txt
```

Ejemplo, con `entrada.txt`:

```
uno,dos,      ,785
+*@#$a3  ultimo
```

La salida del programa debiera ser (más allá de detalles estéticos):

```
Cadena: uno
Separador: ,
Cadena: dos
Separador: ,
Separador: ,
Cadena: 785
Cadena: +*@#$a3
Cadena: ultimo
Fin De Texto:
```

Las entregas serán válidas entre el viernes 6 de agosto y el viernes 27 de agosto.

Si bien hay tiempo recomendando hacerlo lo antes posible para que si lo que entregan no llega a estar bien, tengan tiempo para hacer otra entrega con los errores corregidos.



Forma de entrega: via email, desde su cuenta institucional al mio, indicando que ya lo subieron a su repositorio personal en github. En el mail indique nombre, apellido y curso al que pertenecen.