



Trabajo Práctico N° 2 – Escáner

Hecho con Tabla de Transición

1. Objetivos

- Armar un escáner para lenguaje mini.
- Conocer a bajo nivel como trabaja un escáner.

2. Temas

- Reconocedor léxico.
- Autómatas finitos.

3. Tareas

- Se pide armar un escáner para el lenguaje mini que es descripto en la documentación adicional entregada junto con este enunciado. Para ello debe usar una tabla de transición y basarse en el “seudocódigo para TT” de la presentación dada en clase.
Se debe programar un archivo `main` que pruebe el escáner, llamándolo hasta que el token devuelto sea FDT (EOF).
El escáner debe trabajar contra `stdin` y `stdout`. Luego para probar podremos redireccionar el archivo `entrada.txt` al flujo `stdin`. Se provee el archivo `entrada.txt` para que puedan probar. Debería dar una salida similar a la mostrada en el archivo `salida.txt`
- Programar usando los siguientes fuentes
 - `main.c` llama al escáner hasta que este devuelve el token FDT. Con cada devolución del escáner informa en `stdout` que token fue reconocido y el lexema correspondiente.
 - `scanner.c` tiene el escáner propiamente dicho y sus funciones auxiliares, por ejemplo, una que en base a un carácter devuelva el índice de la columna a usar en la tabla de transición.
 - `scanner.h` tiene la información necesaria que debe conocerse en otros fuentes para utilizar el scanner.
- Arme antes de programar la documentación de la tabla de transición a usar, puede usar una planilla y entregar ese archivo, o exportarlo a pdf o usar un procesador. Puede documentar entregando en cualquier formato en tanto no utilice formatos propietarios.
- Puede usar un `makefile` si lo considera útil, pero no es requerido.
- Si bien un scanner no suele entregar tokens de error, en este caso si lo haremos de modo que `main` sirva para probar todo.
- Siendo que este escáner lo hacemos “a mano” haremos algunas simplificaciones y cambios con respecto a la documentación
 - Para esta versión no hay comentarios que eliminar
 - No reconoceremos las palabras reservadas, las mismas se informarán como identificadores.
 - Informaremos 3 tipos de errores
 - Error común o general, tal como está en la documentación
 - Error de asignación, separado en dos casos
 - Se leyó : pero luego no vino =
 - Aparece un = sin que inmediatamente antes se leyera :



4. Productos

```
`24-002-xx                                //Repositorio del grupo
|-- readme.md                             // Carátula del grupo, ya hecha en TP1
`-- TP2                                   //Directorio para el TP2
    |-- readme.md                         // Carátula del TP
    |-- Tabla.???                        // Documentación de la tabla de transición
    |-- main.c                           // Inicio del programa
    |-- scanner.c                        // Reconocedor léxico
    |-- scanner.h                        // Interfaz del anterior
```

5. Fechas de entrega

- a. Última fecha para primera entrega: 11/09/2024
- b. Última fecha para segunda entrega: 25/09/2024