



# **TÉCNICAS DE COMPILACIÓN**

**TRABAJO PRÁCTICO Nº1**

**ALUMNO: Johana Testa**

## Introducción

---

Dado un archivo de entrada en lenguaje C, se debe generar como salida el Árbol Sintáctico (ANTLR) correcto y mostrar por consola o archivo el contenido de la Tabla de Símbolos de cada contexto. Esto es válido únicamente cuando el archivo de entrada es correcto. Para esta versión se deberá realizar un control de errores básicos y generar un reporte de lo encontrado para poder analizar archivos con posibles errores de codificación. El reporte de errores podrá realizarse por consola o archivo de texto.

Los errores a detectar deben ser los siguientes:

Errores sintácticos comunes:

- Falta de un punto y coma,
- Falta de apertura de paréntesis,
- Formato incorrecto en lista de declaración de variables

Errores semánticos comunes:

- Doble declaración del mismo identificador,
- Uso de un identificador no declarado,
- Uso de un identificador sin inicializar,
- Identificador declarado, pero no usado,
- Tipos de datos incompatibles

## Desarrollo

---

En primer lugar, Se planteó lo que ya se tenía del tp1 y se empezó a escribir un listener personaliza que devolviera los problemas que se podían llegar a obtener como lo son la falta de un paréntesis, un punto y coma o una asignación errónea.

Para realizar las pruebas correspondientes con cada regla, cree archivos `.txt` conteniendo casos de prueba con ejemplos de la estructura a controlar y luego correr con la regla correspondiente. Al finalizar el control de todas las reglas individualmente, hice un caso de prueba conteniendo todos los otros casos testeados anteriormente, y con la estructura de un programa C.

## Conclusión

---

La gramática de un lenguaje de programación ya que se tiene que tener en cuenta los procesos y procedimientos que el mismo debe cumplir. Además, es muy importante tener en cuenta el orden de los TOKENS ya que esto puede hacer que la interpretación cambie y sea errónea, así como también hay que tener en cuenta la dependencia de las reglas gramaticales para evitar recursiones indeseadas.