

[M3] Fase de Compilación: Generador de Analizadores Sintácticos

Integrantes:

- Gerardo Pineda 22880
- Marlon Hernández 15177

Repositorio: https://github.com/Gerax5/Sintactic_gen

Este proyecto implementa **un generador autónomo de analizadores léxicos (YALex) y sintácticos SLR(1) (YAPar)** acompañado de una **interfaz gráfica** para facilitar su uso. A partir de especificaciones formales, la herramienta produce:

- lexer.py – analizador léxico (DFA) listo para usar.
- sintactic.py – analizador sintáctico SLR(1) independiente.
- automata_LR0.png – visualización del autómata LR(0) generado.

Con estos archivos puede **evaluar archivos de prueba** y obtener tokens, diagnóstico de errores léxicos y errores/aceptación sintáctica.

Diseño

1. Generador Léxico

1. **Parseo YALex** → clase Lexer (módulo auxiliar).
2. Procesamiento de **let** para macro-expansión.
3. **Expresión regular global** mediante `pre_process_regex`.
4. **Construcción del AFD** (`Create_automata`).
5. **Serialización** de transiciones, alfabeto y acciones en `lexer.py`.

2. Generador Sintáctico

6. **Lectura YAPar** → tokens declarados, producciones, símbolo inicial.
7. **Verificación** de tokens con el lexer (opcional -l).
8. **Cálculo FIRST / FOLLOW** (First, Follow).
9. **Autómata LR(0)** (AutomataLR0) y graficación (`graph` → PNG).
10. **Tablas SLR(1)** (SLR):

- a. ACTION – desplazamiento, reducción, aceptación.
 - b. GOTO – transiciones con no-terminales.
11. **Código autónomo** (sintactic.py) con algoritmo de pila canónica.

3. Interfaz Gráfica

12. **Selección** de archivos y nombres de salida.
13. Ejecución de los generadores como sub-procesos.
14. **Carga dinámica** de lexer.py / sintactic.py para evaluar cadenas.
15. Panel de texto con color para errores y **mini-visor** del autómata LR(0).
16. Evaluar cadenas de entrada y señalar errores sintácticos.