



UNIVERSIDAD AUTÓNOMA METROPOLITANA
División de Ciencias Naturales e Ingeniería
Departamento de Matemáticas Aplicadas y Sistemas
Licenciatura en Ingeniería en Computación

Traductores 23-P

Proyecto: Una Traductor para el Lenguaje Pantera construido con ANTLR.

Alumnos: *Sara Fuentes Eduardo Christian Trejo Delgado*

Lenguaje Pantera

Para la construcción del traductor del lenguaje Pantera con ANTLR, es necesario conocer la estructura y papel que juega en todo este proceso. En primer lugar ANTLR es un generador de Analizadores sintácticos (**Parsers**) y léxicos (**Lexers**) para construir intérpretes o compiladores para el procesamiento del lenguaje. Su funcionamiento se logra en diferentes etapas en donde analiza y procesa el lenguaje especificado.

En primer lugar, debemos definir la gramática (que consta de un cuarteto)

[

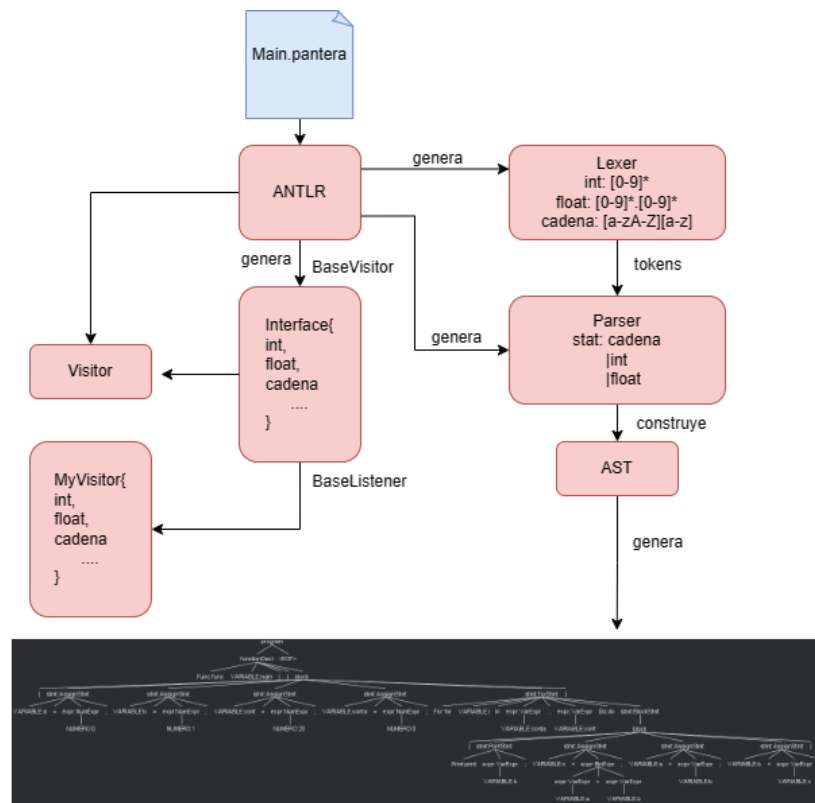
- Símbolo inicial.
- Símbolos Terminales (definidos en términos de Expresiones Regulares)
- Símbolos No Terminales (Definidos en términos de reglas de producción)
- Reglas de producción (se refieren a las especificaciones que describen la sintaxis de un lenguaje).

]

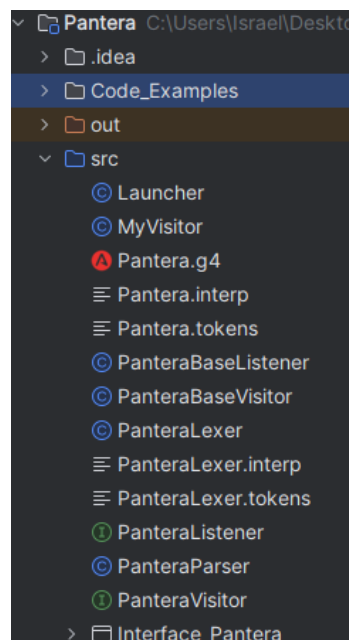
Gracias a esta gramática podemos describir la estructura y sintaxis de nuestro lenguaje, en este caso “Pantera” y ANTLR genera el **analizador léxico (lexer)** cuya principal función es tomar nuestro flujo de texto como entrada y dividirlo en partes más pequeñas llamadas tokens, los cuales son los componentes básicos de nuestro lenguaje i.e las palabras reservadas como (while, for, if, else, def .. etc), identificadores, tipos de datos, operaciones etc, el analizador léxico se encarga de reconocerlos en base a las reglas de producción de la gramática de nuestro lenguaje.

Con la serie de tokens obtenidos por el analizador léxico, sirven como entrada al **analizador sintáctico (Parser)** el cual verifica si los tokens cumplen con las reglas de producción para conocer si el programa cumple con la sintaxis del lenguaje y a partir de eso construir un árbol de sintaxis (AST) que representa la estructura del lenguaje de acuerdo a la gramática, en donde los nodos del árbol son las reglas y tokens definidos en la gramática.

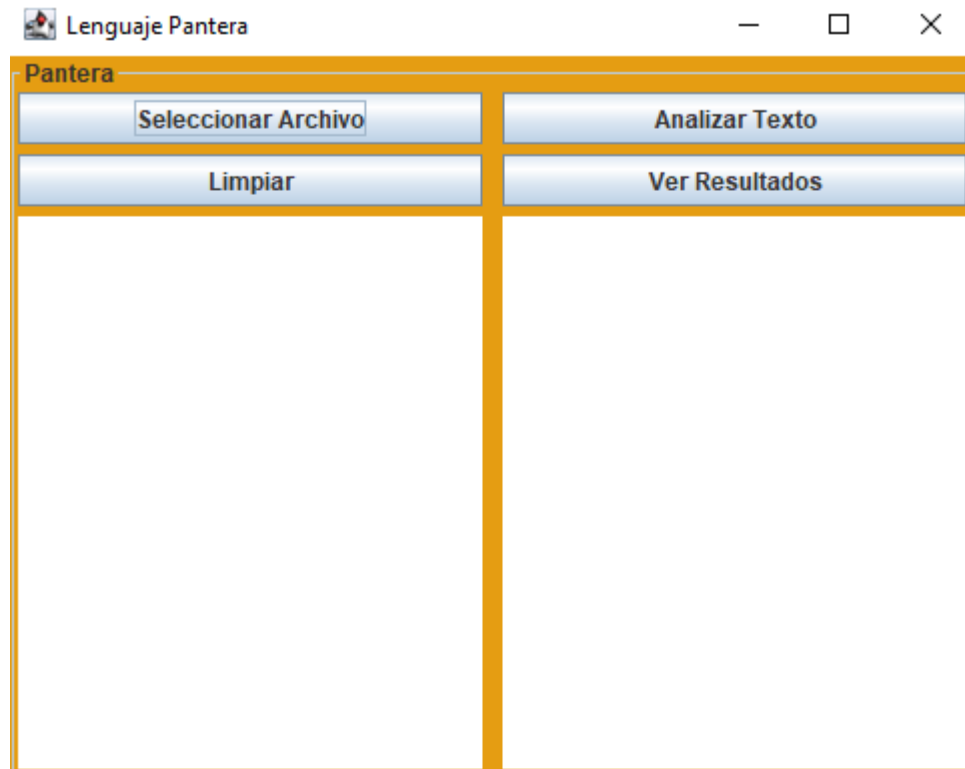
En la siguiente figura se muestra un esquema sobre la estructura que ANTLR usa para construir el lenguaje Pantera.



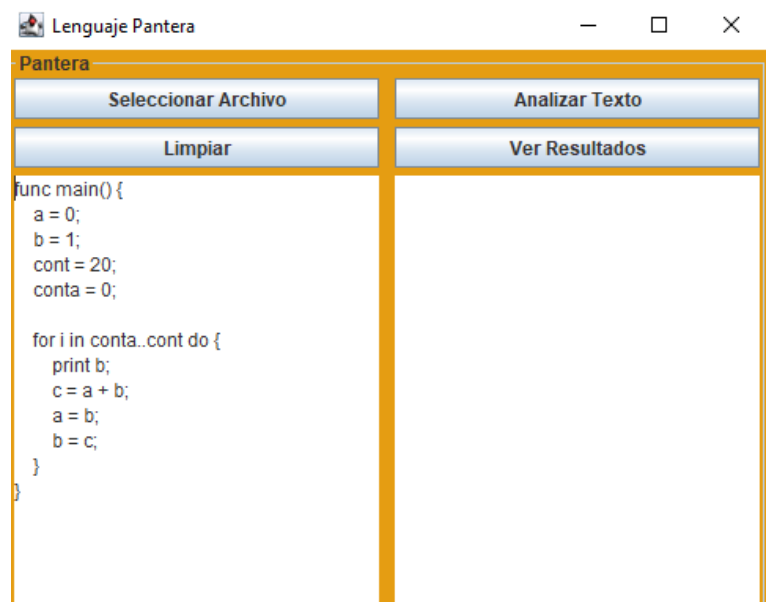
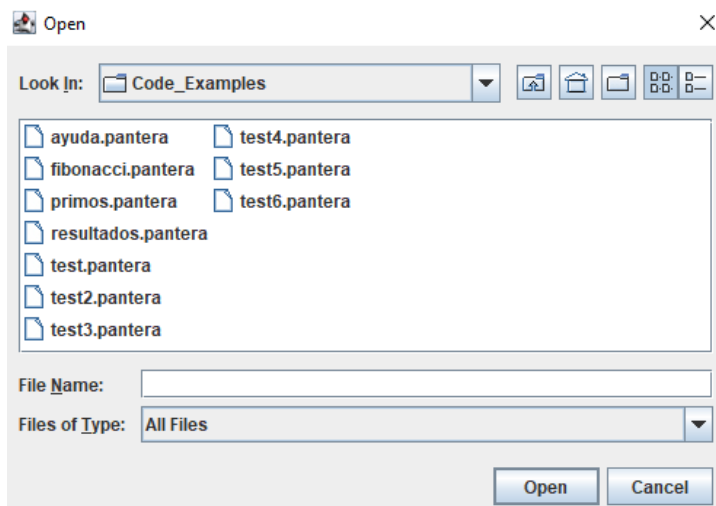
Con el árbol generado por el parser, podemos recorrerlo usando oyentes y visitadores que nos permitan interpretar las instrucciones de nuestro programa. La clase Visitador es generada por ANTLR y se trata de una *Interface* en donde se encuentran definidas todas nuestras reglas de producción y que nos permite definir operaciones sobre los nodos del árbol de sintaxis. Para ejecutar el código perteneciente a una regla en especial se hace con el método **visit()** de ese nodo. Cabe mencionar que el **Visitor()** se basa en otra clase que se llama **BaseVisitor()** que proporciona una implementación por defecto de los métodos **visit** para recorrer y operar sobre los nodos de un árbol de sintaxis (AST). A continuación se muestra los archivos que genera ANTLR:



Para apreciar de mejor manera los resultados y ejemplos de nuestro lenguaje implementamos una pequeña interfaz gráfica como se aprecia en la siguiente figura.



En esta interfaz, el usuario puede elegir un archivo de su directorio que sea de tipo Pantera o en este caso ya se tienen archivos predeterminados que contienen algunos ejemplos de la estructura de un programa en lenguaje Pantera, una vez que el usuario lo selecciona el contenido del archivo se vacía en un área de texto.



Posteriormente el usuario debe de presionar el botón de “Analizar Texto” para que ANTLR comience su análisis y muestre los resultados en la otra área de texto, cabe señalar que para que el usuario sepa que ya termino el análisis se despliega un mensaje que indica la terminación del análisis.

