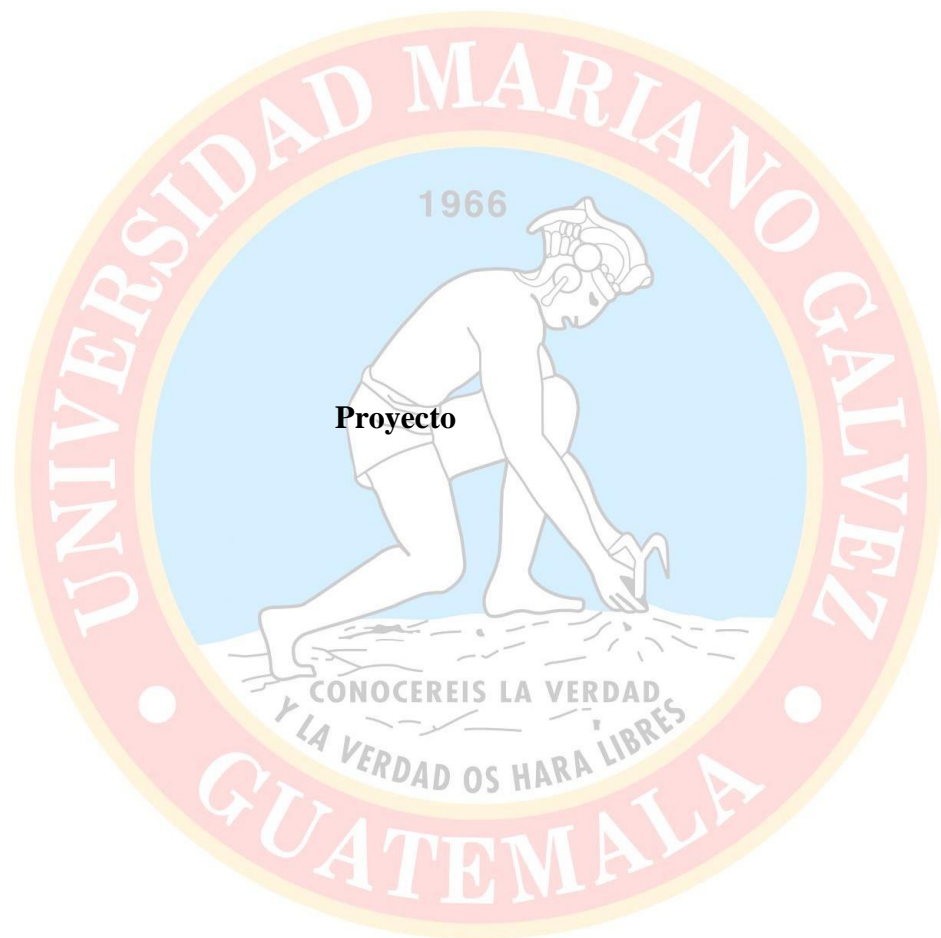


Universidad Mariano Gálvez de Guatemala
Centro Universitario Boca del Monte

Facultad de ingeniería en Sistemas
Ciclo VII



Elmer Fernando Monterroso Quisquinay 7690-20-16343

Guatemala 7 de mayo de 2023

Introducción

La creación de un procesador de lenguaje es una tarea esencial en la computación. Ya sea para interpretar comandos en un terminal, analizar y comprender código escrito en un lenguaje de programación, o para entender y responder a consultas escritas en lenguaje natural, la capacidad de procesar y entender lenguajes es fundamental por lo cual se desarrolla una gramática para java que acepte cierto tipos de subconjunto de producciones.

Justificación

ANTLR es una herramienta potente para generar analizadores léxicos y sintácticos por lo cual se ha utilizado para ver los resultados de nuestra gramática creada y aprender el uso de esta herramienta.

Objetivos

- Aprender la herramienta de ANTLR.
- Probar la herramienta ANTLR.
- Ver en ejemplo real el uso de un analizador léxico y sintáctico.
- Aplicar los conocimientos aplicados en clase.

Análisis y Desarrollo de la aplicación

Para comenzar, definimos una gramática en ANTLR que describe la estructura y las reglas sintácticas del lenguaje Java. En nuestro caso, hemos utilizado la gramática Java.g4 de ANTLR para Java. Esta gramática define reglas para diferentes constructos de Java, como declaraciones de variables, expresiones aritméticas, declaraciones de clases, etc.

Después de generar el código fuente Java a partir de la gramática con ANTLR, escribimos nuestro código Java personalizado para visitar y procesar el árbol de análisis generado por ANTLR. Utilizamos el patrón Visitor para recorrer el árbol y aplicar la lógica correspondiente a cada nodo del árbol.

La gramática acepta

- a) Declaración de variables
 - b) Asignaciones
 - c) Funciones básicas de entrada/salida (print, read, println, ...)
 - d) Expresiones aritméticas y lógicas, y e)
- Estructuras Cíclicas o Repetitivas (while)

Manual

Paso	Descripción
1	Asegúrate de tener instalado el entorno de desarrollo Java en tu computadora. Si no lo tienes, descárgalo e instálalo.
2	Descarga la biblioteca ANTLR y asegúrate de tener el archivo JAR de ANTLR en tu proyecto. Puedes descargarlo desde el sitio oficial de ANTLR.
3	Crea un archivo Java en tu proyecto y asegúrate de importar las clases y paquetes necesarios para utilizar ANTLR.
4	Define la gramática de tu lenguaje utilizando ANTLR en un archivo .g4.
5	Ejecuta ANTLR para generar los archivos Java correspondientes a tu gramática. Utiliza el comando <code>java -jar antlr-4.7-complete.jar <nombre de tu archivo>.g4 -no-listener -visitor -package <nombre del paquete></code> . Esto generará los archivos necesarios para el análisis léxico y sintáctico de tu lenguaje.

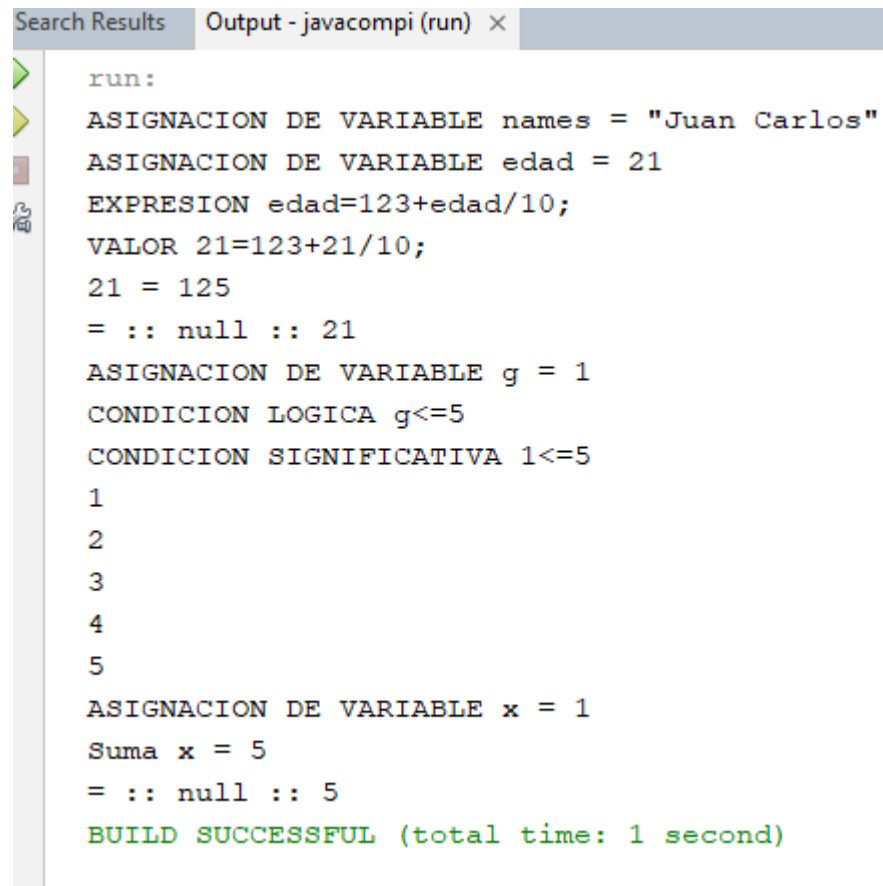
Paso	Descripción
6	En tu archivo Java, crea una clase principal y configura el análisis léxico y sintáctico utilizando ANTLR.
7	Define la lógica para cada tipo de expresión en los métodos correspondientes de tu clase principal.
8	Crea un archivo de texto llamado "input.txt" en la misma carpeta que tu archivo Java. En este archivo, coloca los diferentes casos de inputs que desees probar. Cada caso debe estar en una línea separada.
9	En tu clase principal, lee el contenido del archivo "input.txt" y obtén cada línea como un input de prueba.
10	Para cada input de prueba, realiza el análisis y evaluación correspondiente utilizando ANTLR y tu lógica implementada.
11	Imprime los resultados obtenidos y compáralos con los resultados esperados para cada input.
12	Verifica si los resultados coinciden con los esperados y maneja cualquier error o problema que pueda surgir.
13	Realiza las correcciones necesarias en tu código y vuelve a ejecutar el proyecto para verificar si los resultados son los esperados.
14	Repite los pasos 10 a 13 con diferentes inputs de prueba para probar diferentes casos y asegurarte de que tu programa funcione correctamente.
15	Una vez que hayas probado diferentes inputs y verificado que tu programa funciona correctamente, estará listo para su uso.

Pruebas de ejecución:

Entrada:

```
public class Main {  
    public static void main(String[] args) {  
        String names = "Juan Carlos";  
        int edad = 21;  
  
        edad = 123 + edad / 10;  
        System.out.println("edad: " + edad);  
  
        int g = 1;  
        while (g <= 5) {  
            System.out.println(n++);  
            n++;  
        }  
  
        int x = 1;  
        x = x + 4;  
        System.out.println(x);  
    }  
}
```

Salida:



```
Search Results  Output - javacompil (run) ×  
  
run:  
ASIGNACION DE VARIABLE names = "Juan Carlos"  
ASIGNACION DE VARIABLE edad = 21  
EXPRESION edad=123+edad/10;  
VALOR 21=123+21/10;  
21 = 125  
= :: null :: 21  
ASIGNACION DE VARIABLE g = 1  
CONDICION LOGICA g<=5  
CONDICION SIGNIFICATIVA 1<=5  
1  
2  
3  
4  
5  
ASIGNACION DE VARIABLE x = 1  
Suma x = 5  
= :: null :: 5  
BUILD SUCCESSFUL (total time: 1 second)
```

Video con ampliación de explicación

<https://youtu.be/52cSIVG0T8A>