

***UNIVERDIDAD TECNICA PARTICULAR DE LOJA***

*La Universidad Católica de Loja*

***Modalidad Presencial***

***Área de Computación***

*Proyecto de Teoría de Autómatas y Compiladores*

“Documentación del *Analizador Léxico “*

***Nombre***

Luis León

***Paralelo B***

**Periodo Académico**

ABRIL/2025 - AGOSTO/2025

1. Descripción General del Proyecto

El proyecto cosiste en la creación de un analizador léxico en un lenguaje de programación de tu preferencia que permita leer un archivo fuente (Identificadores, numeres, operadores, etc.) Y que permita leer resultados en un archivo llamado token. json, luego de leer el archivo permita mostrar de manera visual el resultado del archivo.

2. Componentes del Proyecto

Archivos Java

lexer.java: Este archivo es el motor de nuestro código ya que con el nos permite leer línea por línea el archivo fuente que es programa.txt para así poder analizar los tokens mediante expresiones y guardarlos en formato JSON.

token. json: En esta clase definimos la estructura que tendrá cada token, con su contenido y valor el cual se va a generar automáticamente con los tokens recibidos.

programa.txt: este archivo contiene el código fuente el cual se va a analizar léxicamente.

2.2. Visualizador Web

Index.html: Este archivo utiliza la biblioteca Raphaej.js para luego presentar de manera visual un automáticamente con los tokens procesados.

2.2. Visualizador Web (HTML + JavaScript)

index.html: Archivo HTML que utiliza la biblioteca Raphael.js para representar gráficamente un autómata con los tokens procesados.

3.Definición de los lenguajes

3.1 Estructura

Este proyecto acepta expresiones y asignaciones básicas como:

Identificador=expresión;

3.1. Estructura Aceptada

El lenguaje definido acepta expresiones y asignaciones en formato sencillo como:

identificador = expresion;

3.2. Sintaxis del Token

ID: [a-zA-Z\_][a-zA-Z0-9\_]\*

NUM: \d+(\.\d+)?

REL\_OP: >=|<=|==|!=|>|<

ASSIGN: =

END: ;

OP: +|-|\*|/

LPAREN: (

RPAREN: )

4. Tecnologías Utilizadas

En este proyecto se utilizó las siguientes tecnológicas para poder llegar al resultado final.

Java 17: Para desarrollar el analizador léxico.

HTML + JavaScript: Para la presentación de manera visual en el navegador.

Raphael.js: Librería grafica para dibujar el autómata.

5. Funcionamiento

En la consola se ejecuta el lexer.java el cual tiene el problema a analizar en programa.txt línea por línea, luego los tokens validos se guardan en un archivo token. json, y con eso solo quedaría abrir el HTML el cual va a mostrar el resultado con los tipos de tokens y sus valores procesados.