

## Documentación del Analizador Léxico en Flex

### Introducción

Este documento describe el funcionamiento y la implementación de un analizador léxico utilizando Flex. El objetivo de este analizador es reconocer distintos elementos léxicos de un lenguaje de programación, como operadores, números, identificadores, paréntesis, llaves, cadenas de texto y comentarios.

### Archivos Utilizados

**lexer.l:** Archivo fuente de Flex que define las reglas de análisis léxico.

**lexer.h:** Archivo de cabecera con las declaraciones necesarias

### Dependencias

Para compilar y ejecutar este analizador léxico, es necesario tener instalado:

Flex (Fast Lexical Analyzer)

Un compilador de C (como GCC)

### Estructura del Archivo lexer.l

El archivo del analizador está dividido en tres secciones:

1-Definiciones (%{ ... %}): Incluye cabeceras de C y archivos adicionales necesarios.

2-Reglas léxicas (%% ... %%): Contiene las expresiones regulares y las acciones asociadas.

3-Código adicional: Funciones auxiliares, como yywrap() y main().

### Reglas Léxicas

Expresión Regular	Descripción
\[	Corchete de apertura
\]	Corchete de cierre
\(	Paréntesis de apertura
\)	Paréntesis de cierre
[0-9]+(\.[0-9]+)?	Números enteros y decimales
[+/*=]	Operadores aritméticos
[a-zA-Z_][a-zA-Z0-9_]*	Identificadores (variables, funciones)
\("[^"]*"	Cadenas de texto entre comillas dobles
'[^']*'	Caracteres literales
] *`	Comentarios de una línea (//)
\[/*[^\n]*\n*/\]	Comentarios multilinea (/ * ... */)
[ \t\n]	Espacios y saltos de línea (ignorados)

;	Punto y coma
\{	Llave de apertura
\}	Llave de cierre
.	Carácter no reconocido

### Compilación y Ejecución

#### Generar el código en C a partir del archivo Flex:

```
flex lexer.l
```

#### Compilar el código generado con GCC:

```
gcc lex.yy.c -o lexer -lfl
```

#### Ejecutar el analizador:

```
./lexer < archivo_de_prueba.txt
```

### Conclusión

Este analizador léxico en Flex permite identificar y clasificar elementos básicos de un lenguaje de programación. Puede ser ampliado para reconocer nuevas estructuras y adaptarse a necesidades específicas.