

Primer programa
Analizador Léxico

Objetivo

Elaborar un analizador léxico en *lex/flex* que reconozca los componentes léxicos pertenecientes a las clases abajo descritas.

Descripción

- Las clases de los componentes léxicos válidos para el analizador léxico son:

Clase	Descripción
0	Constantes numéricas reales Ej: 2.17, 0.39, 2.7E-3, 3.5E5, 18.0e+5
1	Palabras reservadas (ver tabla)
2	Símbolos especiales { } , : () []
3	Identificadores. Letra mayúscula o minúscula y puede estar seguida de letras mayúsculas o minúsculas, dígitos o guiones bajos.
4	Operador de asignación =
5	Operadores relacionales (ver tabla)
6	Constantes numéricas enteras (base 10, sin sufijos y hasta 14 dígitos)
7	Cadenas. Cualquier número de caracteres encerrados entre comillas (""), incluye cadena vacía.
8	Operadores aritméticos + - * / ** %

El número de clase es inamovible.

- El analizador léxico tendrá como entrada un archivo con el programa fuente, el cual se indicará desde la línea de comandos al momento de mandar a ejecutar el analizador léxico.
- Como delimitador de un componente léxico será uno o varios espacios, tabuladores o saltos de línea, **así como el inicio de otro componente léxico.**
- Los tokens se representarán en una estructura con dos campos:
campo1: la clase
campo2: el valor (de acuerdo a las siguientes tablas e indicaciones)

Valor	Palabra reservada
0	bool
1	break
2	case
3	char
4	continue
5	default
6	do
7	else
8	float

Valor	Palabra reservada
9	for
10	if
11	int
12	return
13	string
14	switch
15	while
16	void

Valor	Op. relacional
0	!=
1	==
2	>
3	<
4	>=
5	<=

El valor para el token de cada identificador es la posición dentro de la tabla de símbolos. Para los operadores relacionales y las palabras reservadas, el valor del token será la posición en su correspondiente tabla (catálogo).

Las constantes numéricas reales y las cadenas, tendrán cada una su tabla de literales; por lo que el valor de su token será la posición dentro de la tabla de literales correspondiente. Cada que el Analizador Léxico encuentre una cadena o número real, NO revisará si ya se encuentra en su tabla de literales, simplemente la insertará.

Para las clases 2 y 4, el valor en su token es el mismo carácter o su correspondiente ASCII

Con respecto a la clase 8 (operadores aritméticos), el valor del token será el carácter +, -, *, / o %, según corresponda. Para el operador **, el valor en su token será 0 (cero).

- Deberá reconocer y descartar comentarios que están definidos por iniciar con @ y hasta el final de línea
- Cuando detecte un error léxico, deberá seguir el reconocimiento a partir del siguiente símbolo válido.
- El analizador deberá crear la Tabla de Símbolos para almacenar a los identificadores. Esta tabla manejará los campos: posición, nombre del identificador y tipo (este último será de tipo entero y podrán ponerle como valor inicial -1). Se indicará en el documento a entregar, el tipo de estructura de datos empleada, así como el método de búsqueda a utilizar.
- Las Tablas de Literales, una para los reales y otra para las cadenas, deberán tener como estructura, dos campos: la posición y el dato (cadena o número real) según el caso.
- Al término del análisis léxico deberá mostrar la tabla de símbolos, las tablas de literales, así como los tokens. También podrán almacenarse en archivos para su mejor revisión.
- Los errores que vaya encontrando el analizador léxico, los podrá ir mostrando en pantalla o escribirlos en un archivo. Deberá recuperarse de los errores encontrados para continuar con el reconocimiento de todos los componentes léxicos.
- El programa deberá estar comentado, con una descripción breve de lo que hace (puede ser el objetivo indicado en este documento), el nombre de quién(es) elaboró(aron) el programa y fecha de elaboración, así como lo que hace cada función. Se deberá cuidar mucho la sangría que denota la dependencia de instrucciones.

Entregar:

Un documento con la siguiente estructura:

- Descripción del problema (no del programa), incluyendo características que deben cumplir los componentes léxicos que reconocerá el analizador léxico y la expresión regular de cada clase.
- Propuesta de solución y fases del desarrollo del sistema: Análisis (planificación, indicando participantes por cada actividad), Diseño e Implementación. En el diseño indicar exactamente cómo se definirá la tabla de símbolos, las tablas de literales y los tokens, sus técnicas de búsqueda e inserción.
- Indicaciones de cómo correr el programa.
- Conclusiones por cada participante.

Nota: se podrá elaborar individualmente o en equipo de 2

Enviar el documento y sólo el programa fuente definitivo a la plataforma educativa y en su caso, sólo un miembro del equipo.

Fecha de entrega: 29 de octubre de 2020.