

Analizador léxico con ANTLR Hoja de ejercicios



Procesadores de Lenguajes

Jaime Urquiza Fuentes

Listado de problemas sobre desarrollo de analizadores léxicos con ANTLR



Ejercicios de traductores léxicos con ANTLR

Diseñar una especificación ANTLR que:

1. Dado un fichero de texto de entrada, cambie todas las vocales minúsculas por vocales mayúsculas.
2. Copie el archivo de entrada en uno de salida, sumando 5 a todo número positivo que sea múltiplo de 3.
3. Dado un fichero de texto de entrada con letras mayúsculas y minúsculas, signos de puntuación y saltos de línea, codifique cada palabra utilizando el siguiente código:
 - Si la palabra termina en vocal, se cambian todas sus letras por la letra anterior en el alfabeto.
 - Si la palabra termina en consonante, se cambian todas sus letras por la letra posterior en el alfabeto.
 - El resto de símbolos se dejan igual.
 - NOTA: después de la “z” se entiende que va la “a”. Se recomienda usar la clase Carácter de Java.
4. Dado un fichero ofrezca como salida un fichero en el que se han eliminado de la entrada las líneas en blanco, los espacios en blanco al principio o al final de línea y que tan sólo deja un blanco entre cada dos palabras consecutivas.
5. Hacer un programa que interprete rangos de números de página. Para la entrada: 1-4;5;8;45-47;60 debería generar: 1,2,3,4,5,8,45,46,47,60
6. Dado un fichero de texto de entrada lo procese de la siguiente forma:
 - Si al comienzo de la línea se encuentra una “m”, cambiará todas las letras mayúsculas por minúsculas.
 - Si al comienzo de la línea se encuentra una “M”, cambiará todas las letras minúsculas por mayúsculas.
 - En otro caso deja la línea como está.
7. Dado un fichero de texto de entrada, genere el mismo fichero con las líneas numeradas:

Entrada	Salida
Primera línea	1:Primera línea
Segunda línea	2:Segunda línea
	3:
Cuarta línea, la 3ª está vacía	4:Cuarta línea, la 3ª está vacía

8. Dado un fichero donde hay constantes numéricas en las siguientes bases:
 - Binaria: “b” seguido de una ristra de dígitos binarios.
 - Octal: “x” seguido de una ristra de dígitos octales.
 - Decimal: ristra de dígitos decimales) y
 - Hexadecimal: “0x” seguido de una ristra de dígitos hexadecimales).

Donde las constantes numéricas pueden ser enteras o reales con signo opcional. Las constantes reales se pueden expresar en formato de punto fijo o coma flotante. A continuación, se muestran algunos ejemplos.

	binaria	octal	decimal	hexadecimal
Entero	-b011	+x56	098	-0x34A5
Real punto fijo	b01.11	x54.01	-83.91	+0xff.A4
Real coma flotante	+b.001e-111	x.74e230	+54e-4	0x.4Ae+FF

NOTAS:

- Como funciones de conversión sólo se permite usar las de las clases Character e Integer.
- Los dígitos hexadecimales no numéricos son siempre letras mayúsculas.
- El exponente de las constantes reales en coma flotante se empieza siempre con “e”.

Se pide devolver los tokens correspondientes: <token, valor numérico en base decimal>