

## Trabajo 2. Analizador lexical

La asignación 1 es construir un lexer (analizador lexical) para el lenguaje Tiger usando ML-Lex. El lenguaje del tigre se define en el Apéndice A del libro de texto del curso. (Se hará llegar por esta vía) La documentación sobre ML-Lex estará disponible y también (en menor detalle) en el capítulo dos del libro de texto.

De inicio: <http://www.cs.tufts.edu/comp/181/tiger.html>

Los archivos de esqueleto para empezar están disponibles en el directorio  
<http://www.cs.princeton.edu/~appel/modern/ml/chap2/>

Te recuerdo que las palabras reservadas del lenguaje son:

while, for, to, break, let, in, end, function, var, type, array, if, then, else, do, of, nil

Separadores:

, : ; ( ) [ ] { } . + - \* / = < > <= >= & | :=

No hay literales enteros negativos

Dentro de un string hay valores de escape

Hay comentarios

Se deben detectar strings no cerrados y comentarios no cerrados

El archivo cm.cm debe ser modificado

Sustituye la línea

smlnj-lib.cm

por

\$/smlnj-lib.cm

Mejor es cambiar todo el cm.cm por

Group is

driver.sml

errmsg.sml

tokens.sig

tokens.sml

tiger.lex: mllex

\$/smlnj-lib.cm

\$/basis.cm

\$/ml-yacc-lib.cm

1.- El producto final tuyo es tiger.lex  
Una vez completado el mismo ejecutarás  
sml  
- CM.make("cm.cm");

También se podría probar con (línea de comando)  
sml -m cm.cm  
o  
sml cm.cm

Y caes en el repl de sml

a.- Lo anterior generarar una salida tiger.lex.sml

2.- Tomarás uno de los programas de la pagina  
<http://www.cs.princeton.edu/~appel/modern/testcases/>  
y lo someterás a  
- Parse.parse "test4.tig" ;  
con salida que quiero ver

3.- Modificarás este u otro programa -no tengo idea que modificación harás- y también lo someterás a la misma prueba  
con salida que quiero ver

Debes enviar:

Un archivo tiger.lex, con la fuente para su lexer.  
Cualquier otro archivo de origen que escribió para apoyar a su lexer.  
Un archivo de texto que describe  
Los miembros de tu equipo  
Cómo manejas los comentarios  
Cómo manejaste los errores  
Cómo manejaste al final del archivo  
Cualquier otra cosa que usted piensa es de interés sobre su lexer

-----  
Evaluación:

Si el archivo especificado no se encuentra en la ubicación especificada de su cuenta en el plazo, recibirá 0 puntos.

Tiger.lex debe ser legible por vi o se le aplicará una penalización de 10 puntos. (hum ?)

Su sistema debe generar tiger.lex.sml de tiger.lex, compilar todos los archivos de fuentes necesarios y ejecutarse con éxito (sin errores) utilizando sólo la siguiente invocación de función desde dentro del entorno SML o recibirá 0 puntos.

- CM.make "sources.cm";