

## Analizador léxico lex

{Programa}

%{

Instrucción            {CicloHasta}{ Selección}{Aritmética}{CicloMientras}{DefinirVariable}

{Instrucción}\{CicloHasta}{Instrucción}\{Selección}

{Instruccion }\{Aritmética}{Instrucción}\{CicloMientras}

{Instrucción}\{DefinirVariable}

DefinirVariable        ({'Definir'}|{'definir'})\{ NombreVariable}\{Tipo}

variableEnMemoria    {NombreVariable}\{Tipo}

Tipo                    {'Entero'}|{'Decimal'}|{'Cadena'}

NombreVariable        {LetraMayu}\{LetraMayu}|{Cadena}

Operando              [0-9]\*

Operandocompuesto   {signo}\{Operando})|({ OperandoCompuesto})\{Operando})

LetraMayus            [A-Z]

Aritmética            ({Operando}{3})|({Aritmetica}\{Operando}{2})|  
                           ({OperandoCompuesto}\{Operando}\{OperandoCompuesto})|  
                           ({Aritmetica})\{Operando}\{OperandoCompuesto})

Operador              { "+" | "-" | "\*" | "/" | ">" | "<" | "<=" | ">=" | "=" | "==" }

CicloHasta            ({'repetir hasta'}\{DefinirVariable})\;"\({'repetir  
                           hasta'}\{DefinirVariable}\{VariableEnMemoria})\;"\;  
                           ({VariableEnMemoria})\;"\;

OperadorCiclos        {"i="}|{">"}|{"<"}|{"<="}|{">="}|{"=="}

Ciclomientras        {'mientras'}\{CondicionAcomprobar}'repetir'

CondicionAcomprobar {VariableEnMemoria}\{OperadorCiclos}\{VariableEnMemoria}

Selección              ('si'\{CondicionAcomprobar}'hacer'\finsi')| 'si'  
                           \{CondicionAcomprobar}\hacer'\sino '\finsi')

%{

%%

[] { ECHO;}

Instrucción { ECHO;}

DefinirVariable { ECHO;}

variableEnMemoria { ECHO;}

Tipo { ECHO;}

NombreVariable { ECHO;}

Operando { ECHO;}

Operandocompuesto { ECHO;}

LetraMayus { ECHO;}

Aritmética { ECHO;}

Operador { ECHO;}

CicloHasta { ECHO;}

OperadorCiclos { ECHO;}

Ciclomientras { ECHO;}

CondicionAcomprobar { ECHO;}

Selección { ECHO;}

. { printf("<<error>>");}

<<EOF>> {yyterminate();}

%%

Main()

{ yylex (); }