

```

datatype lexresult= DIV | EOF | EOS | ID of string | LPAREN |
                  NUM of int | PLUS | PRINT | RPAREN | SUB | TIMES | EQ
val linenum = ref 1
val error = fn x => TextIO.output(TextIO.stdOut,x ^ "\n")
val eof = fn () => EOF
fun refinc x = (x := !x + 1; !x)

%%

%structure Simplex
alpha=[A-Za-z];
digit=[0-9];
ws = [\ \t];
%%X
\n => (refinc linenum; lex());
{ws}+ => (lex());
"/" => (DIV);
";" => (EOS);
"(" => (LPAREN);
"=" => (EQ);
{digit}+ => (NUM (List.foldr (fn(a,r)=>ord(a)-ord("#0")+10*r) 0 (explode yytext)));
")" => (RPAREN);
"+" => (PLUS);
{alpha}+ => (if yytext="print" then PRINT else ID yytext);
"-" => (SUB);
"*" => (TIMES);
. => (error ("calc: ignoring bad character "^yytext); lex());

```