https://github.com/Nyesteban/FLCD-lab-2

%{

#include <math.h>

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int currentLine = 1;

%}

IDENTIFIER [a-zA-Z\_][a-zA-Z0-9\_]\*

NUMBER\_CONST 0|[+|-]?[1-9][0-9]\*([.][0-9]\*)?|[+|-]?0[.][0-9]\*

STRING\_CONST [\"][0-9a-zA-Z.' ]\*[\"]

%%

"output"|"input"|"if"|"else"|"for"|"integer"|"string"|"return"|"array" {printf("Reserved word: %s\n", yytext);}

"+"|"-"|"\*"|"/"|"<="|">="|"=="|"!"|"<"|">"|"="|"&&"|"||"|"%" {printf("Operator: %s\n", yytext);}

"{"|"}"|"("|")"|"["|"]"|";"|","|"\"" {printf("Separator: %s\n", yytext);}

{IDENTIFIER} {printf("Identifier: %s\n", yytext);}

{NUMBER\_CONST} {printf("Number: %s\n", yytext);}

{STRING\_CONST} {printf("String: %s\n", yytext);}

[0-9][a-zA-Z0-9\_]\* {printf("Illegal identifier\n");}

[+|-]0 {printf("Illegal numeric constant\n");}

[+|-]?[0][0-9]\*([.][0-9]\*)? {printf("Illegal numeric constant\n");}

"{"[^}\n]\*"}" /\* eat up one-line comments \*/

[ \t\n]+ /\* eat up whitespace \*/

. printf("Lexical Error\n",yytext);

%%

main( argc, argv )

int argc;

char \*\*argv;

{

++argv, --argc; /\* skip over program name \*/

if ( argc > 0 )

yyin = fopen( argv[0], "r" );

else

yyin = stdin;

yylex();

}





