Github Link: <https://github.com/916RasnitaRadu/SEM5-FLCD/tree/main/Lab8>

**Lex Specification File:**

%{

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int lines = 1;

%}

%option noyywrap

%option caseless

DIGIT [0-9]

NON\_ZERO\_DIGIT [1-9]

LETTER [a-zA-Z]

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

CHAR \'[a-zA-Z0-9]\'

INTEGER 0|[+|-]?[1-9][0-9]\*

STRING \"[a-zA-Z0-9]\*\"

%%

"if"|"else"|"while"|"for"|"integer"|"string"|"char"|"read"|"print"|"return"|"start"|"arr" {printf("%s - reserved word\n", yytext);}

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

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

{IDENTIFIER} {printf("%s - identifier\n", yytext);}

{INTEGER} {printf("%s - integer\n", yytext);}

{STRING} {printf("%s - string\n",yytext);}

{CHAR} {printf("%s - character\n", yytext);}

[ \t]+ {}

[\n]+ {lines++;}

[0-9][a-zA-Z0-9\_]\* {printf("Illegal identifier at line %d\n", lines);}

[+|-]0 {printf("Illegal numeric constant at line %d\n", lines);}

[+|-]?[0][0-9]\* printf("Illegal numeric constant at line %d\n", lines);}

[\'][a-zA-Z0-9 ]{2,}[\']|[\'][a-zA-Z0-9 ][a-zA-Z0-9 ][\'] {printf("Illegal character constant at line %d\n", lines);}

%%

int main(int argc, char\*\* argv) {

if (argc > 1) {

FILE\* file;

file = fopen(argv[1], "r");

if (!file) {

fprintf(stderr, "Could not open %s!\n", argv[1]);

exit(1);

}

yyin = file;

}

yylex();

return 0;

}

**Demo**

We first run the command

Next, we compile the generated C file



Finally, an executable is created which represents the scanner (lexer). We run this executable on one of the input files.

The result is:

