# EXPERIMENT - 2

Implementation of Lexical Analyzer using Lex Tool.

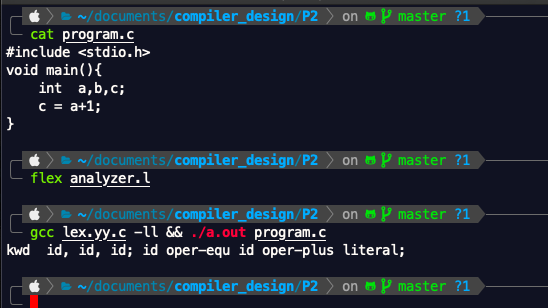
LEX program

|  |
| --- |
| %%  [ \t\n] ; if|else|while|int|float|char|double|for {printf("kwd");} [a-zA-Z][a-zA-Z0-9\_]\* printf("id"); [0-9] printf("literal"); [\+] printf(" oper-plus "); [\-] printf(" oper-minus "); [\/] printf(" oper-divide "); [\\*] printf(" oper-multiply "); [\=] printf(" oper-equ ");  %% int main(int argc, char \*argv) {  FILE \*f;   if (ac == 2) {  if (!(f = fopen(av[1], “r"))) {  perror("Error: ");  return (-1);  }  yyset\_in(f);  yylex();  fclose(f);  }  else  printf("Usage: a.out filename\n");  printf("\n");  return (0); } |
|  |

Input

int a,b,c;

c = a+1;

Output

Readme

1. Run flex analyzer.l
2. Gcc lex.yy.c -ll && ./a.out <file to be analyzed>