26. Write a LEX specification file to take input C program from a .c file and count tthe number of characters, number of lines & number of words.

%{

#include <stdio.h>

int line\_count = 0; // Line counter

int char\_count = 0; // Character counter

int word\_count = 0; // Word counter

%}

%%

\n { line\_count++; } // Increase line count on new line

[ \t]+ { /\* Ignore spaces and tabs in word count \*/ }

[A-Za-z0-9\_]+ { word\_count++; } // Match words (alphanumeric and underscores)

. { char\_count++; } // Count any character (excluding newlines)

%%

int main() {

FILE \*file = fopen("input.c", "r"); // Open the input C file

if (file) {

yyin = file; // Set the input file to be processed by Lex

yylex(); // Start the lexical analysis

fclose(file); // Close the file after processing

} else {

printf("Unable to open the file.\n");

return 1;

}

printf("Number of lines: %d\n", line\_count);

printf("Number of words: %d\n", word\_count);

printf("Number of characters: %d\n", char\_count);

return 0;

}  
27. Write a LEX program to print all the constants in the given C source program file.

%{

#include <stdio.h>

%}

%%

[0-9]+(\.[0-9]+)?[fF]? { printf("Constant: %s\n", yytext); } // Matches integer & floating constants

\"[^\"]\*\" { printf("String literal: %s\n", yytext); } // Matches string literals

%%

int main() {

FILE \*file = fopen("input.c", "r"); // Open the C source file

if (file) {

yyin = file; // Set the input file for Lex

yylex(); // Start the lexical analysis

fclose(file); // Close the file after processing

} else {

printf("Unable to open the file.\n");

return 1;

}

return 0;

}  
28. Write a LEX program to count the number of Macros defined and header files included in the C program.

%{

#include <stdio.h>

int define\_count = 0; // Macro definitions counter

int include\_count = 0; // Header file inclusion counter

%}

%%

^#define[ \t]+[A-Za-z0-9\_]+ { define\_count++; } // Match macro definitions (e.g., #define MACRO\_NAME)

^#include[ \t]+\"[^\"]\*\" { include\_count++; } // Match header file includes (e.g., #include "file.h")

%%

int main() {

FILE \*file = fopen("input.c", "r"); // Open the C source file

if (file) {

yyin = file; // Set the input file for Lex

yylex(); // Start lexical analysis

fclose(file); // Close the file after processing

} else {

printf("Unable to open the file.\n");

return 1;

}

printf("Number of macros defined: %d\n", define\_count);

printf("Number of header files included: %d\n", include\_count);

return 0;

}