NATIONAL INSTITUTE OF TECHNOLOGY SILCHAR

Cachar, Assam

B.Tech. VIth Sem

Subject Code: CS-316

Subject Name: Compiler Design Lab

Submitted By:

Name : Subhojit Ghimire

Sch. Id. : 1912160

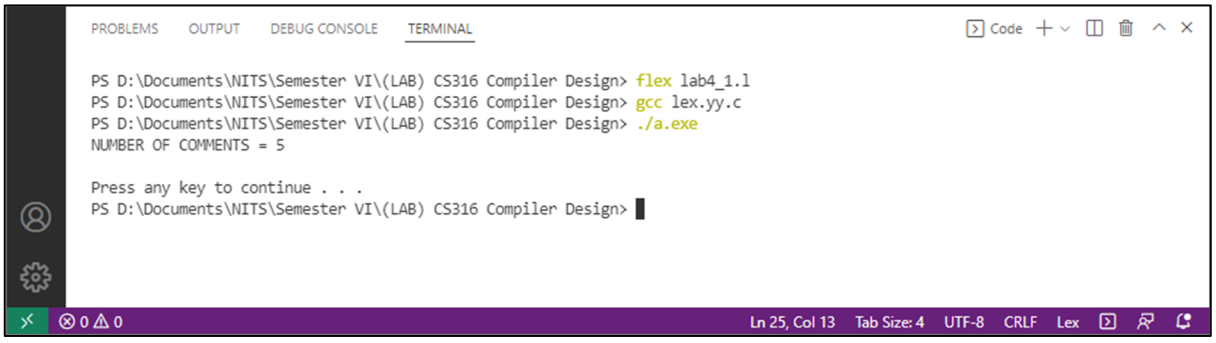
Branch : CSE – B

1. Write a lex program to count the number of comments in a C program, and then delete the comments.

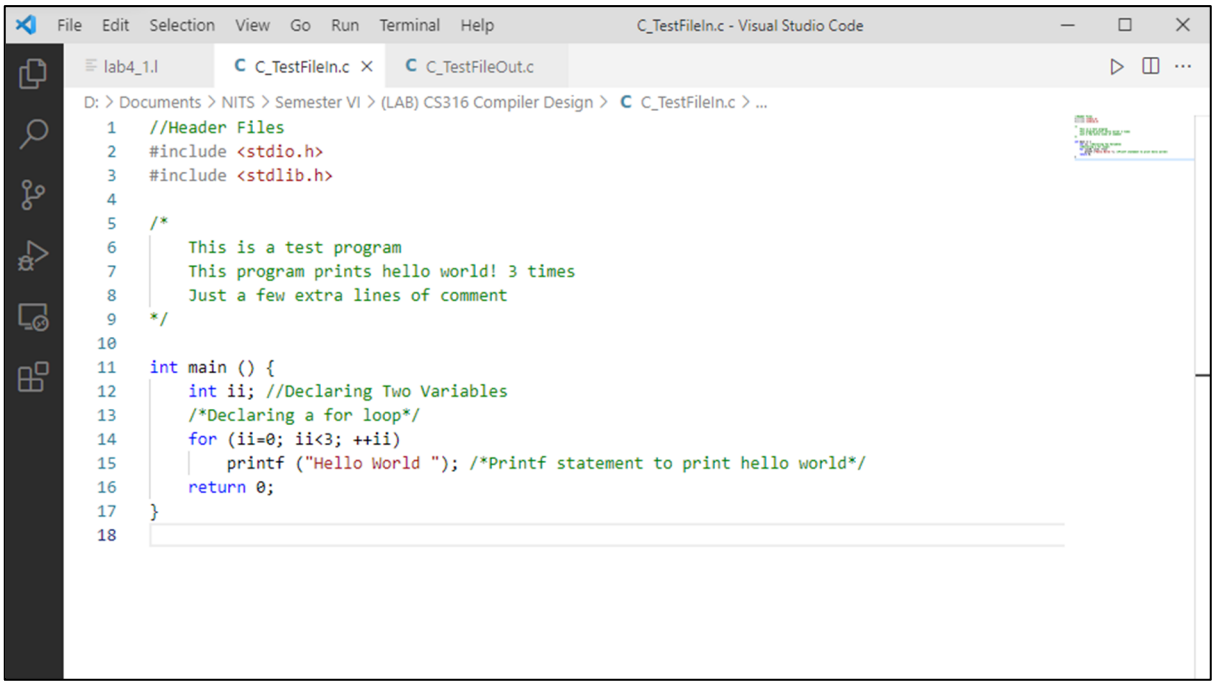
➔ **CODE:**

```
%{
    #include <stdio.h>
    #include <stdlib.h>
    int commentCount = 0;
    FILE *out;
%}
%s commentText

%%
"/*" {BEGIN commentText;}
<commentText>"*/" {BEGIN 0; ++commentCount;}
<commentText>\n {}
<commentText>. {}
"//".* {++commentCount;}
.|\n {fprintf (out, "%s", yytext);}
%%

int yywrap(){
    return 1;
}

int main(){
    yyin = fopen("C_TestFileIn.c", "r");
    out = fopen("C_TestFileOut.c", "w");
    yylex();
    printf ("NUMBER OF COMMENTS = %d\n\n", commentCount);
    fclose (out);
    system ("pause");
    return 0;
}
```

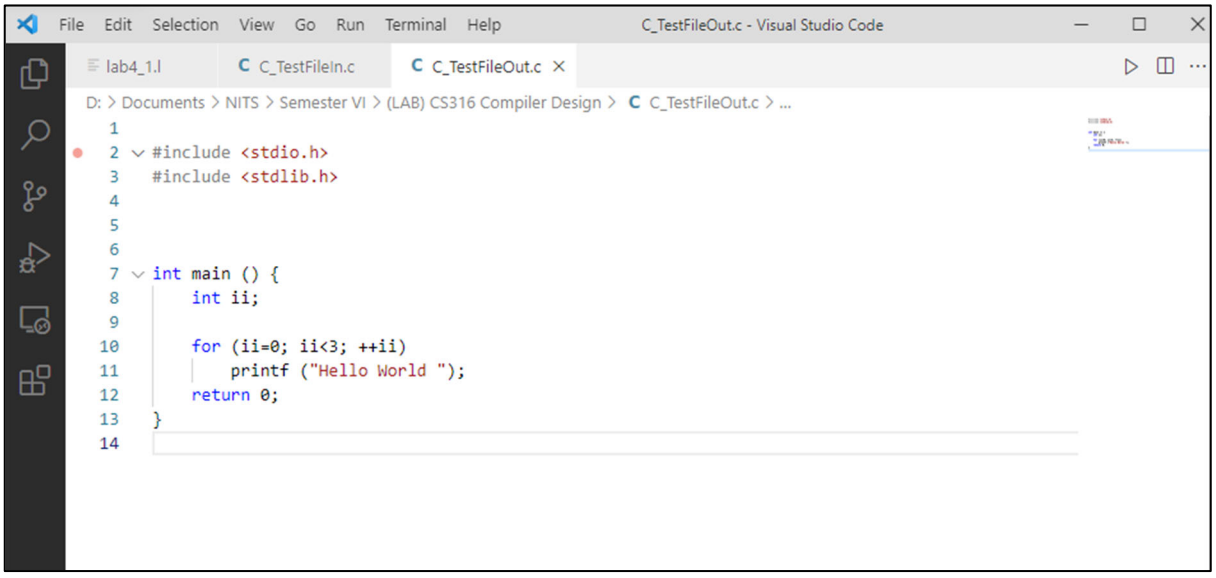
OUTPUT:



// C_TestFileIn.c file



//C_TestFileOut.c file



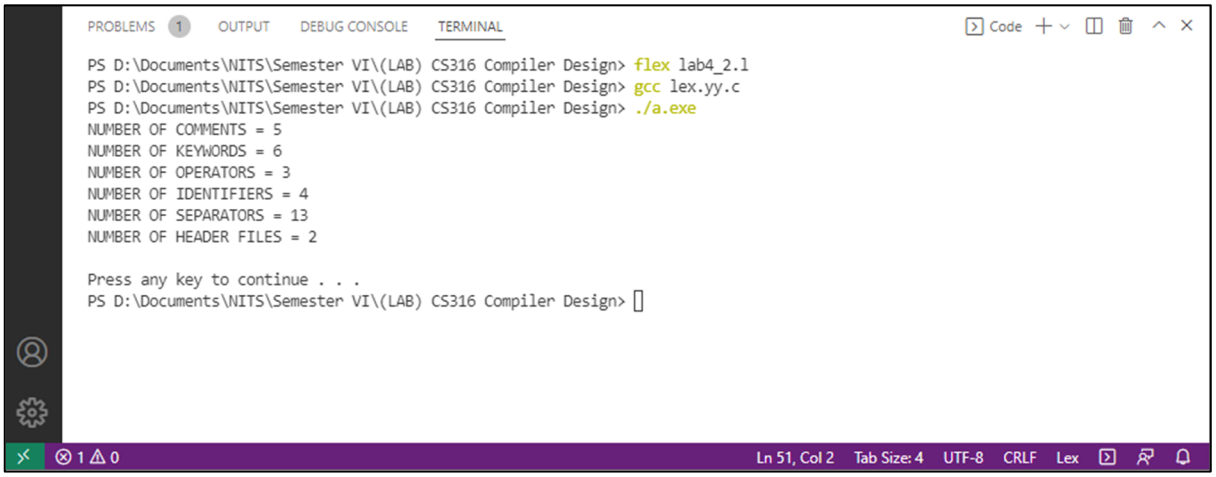
2. Write a lex program to count the number of keywords, operators, identifiers, comments and then delete the comments.

➔ **CODE:**

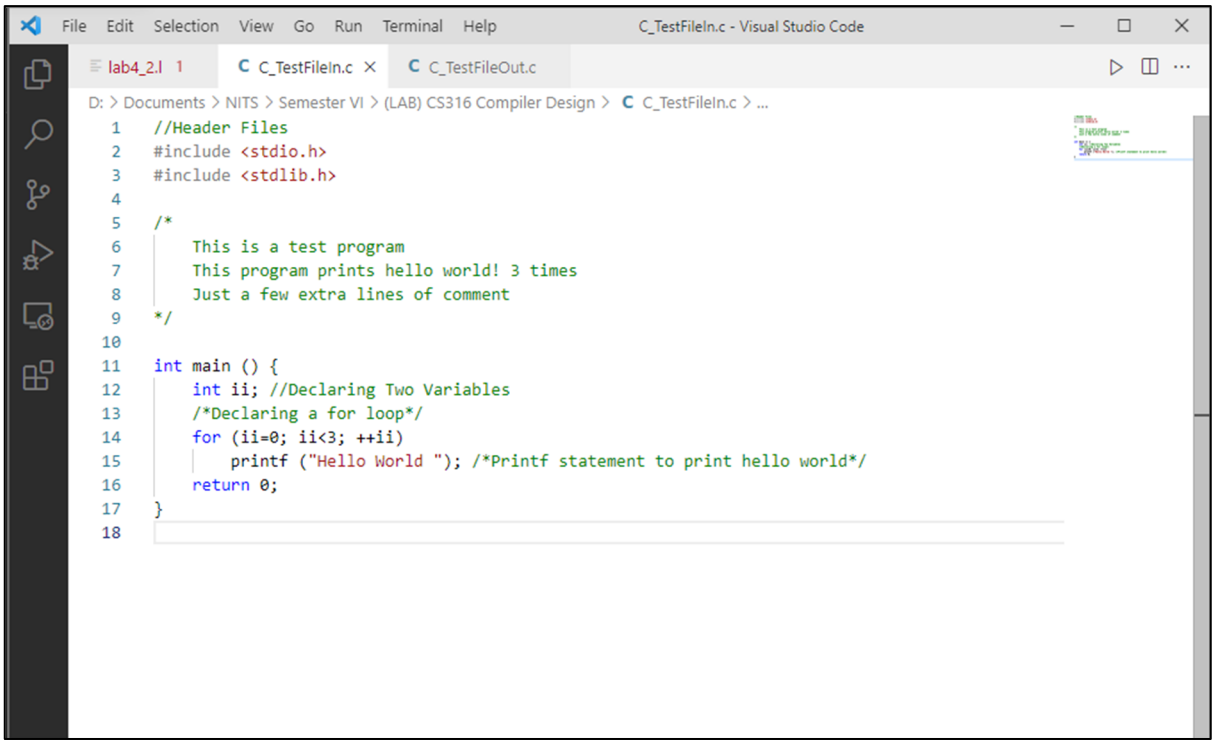
```
%{
    #include <stdio.h>
    #include <stdlib.h>
    int commentCount = 0, keywordCount = 0, operatorCount = 0,
identifierCount = 0;
    int headerfileCount = 0, separatorCount = 0;
    FILE *out;
%}
%s commentText

%%
"/*" {BEGIN commentText;}
<commentText>"*/" {BEGIN 0; ++commentCount;}
<commentText>[a-zA-Z0-9]* {;}
<commentText>\n {;}
<commentText>. {;}
"//".* {++commentCount;}
\n {fprintf (out, "%s", yytext);}
"auto"|"double"|"main"|"int"|"struct" {++keywordCount; fprintf (out,
"%s", yytext);}
"break"|"else"|"long"|"switch"|"case"|"printf" {++keywordCount; fprintf
(out, "%s", yytext);}
"enum"|"register"|"typedef"|"char" {++keywordCount; fprintf (out, "%s",
yytext);}
"for"|"signed"|"void"|"do"|"if" {++keywordCount; fprintf (out, "%s",
yytext);}
"extern"|"return"|"union"|"continue" {++keywordCount; fprintf (out,
"%s", yytext);}
"static"|"while"|"default"|"goto" {++keywordCount; fprintf (out, "%s",
yytext);}
"sizeof"|"volatile"|"const"|"float"|"short" {++keywordCount; fprintf
(out, "%s", yytext);}
"#include"[ \<a-z.A-Z\>]* {++headerfileCount; fprintf (out, "%s",
yytext);}
\".*\" {fprintf (out, "%s", yytext);}
[{;}] {++separatorCount; fprintf (out, "%s", yytext);}
[,()] {++separatorCount; fprintf (out, "%s", yytext);}
[+-/=*%] {++operatorCount; fprintf (out, "%s", yytext);}
[a-zA-Z][a-zA-Z0-9_]* {++identifierCount; fprintf (out, "%s", yytext);}
. {fprintf (out, "%s", yytext);}
%%
```

```c
int yywrap(){
    return 1;
}

int main(){
    yyin = fopen("C_TestFileIn.c", "r");
    out = fopen("C_TestFileOut.c", "w");
    yylex();
    printf ("NUMBER OF COMMENTS = %d\n", commentCount);
    printf ("NUMBER OF KEYWORDS = %d\n", keywordCount);
    printf ("NUMBER OF OPERATORS = %d\n", operatorCount);
    printf ("NUMBER OF IDENTIFIERS = %d\n", identifierCount);
    printf ("NUMBER OF SEPARATORS = %d\n", separatorCount);
    printf ("NUMBER OF HEADER FILES = %d\n\n", headerfileCount);
    fclose (out);
    system ("pause");
    return 0;
}
```

OUTPUT:



// C_TestFileIn.c file



// C_TestFileOut.c file