System Software (CS306)

Assignment - 9

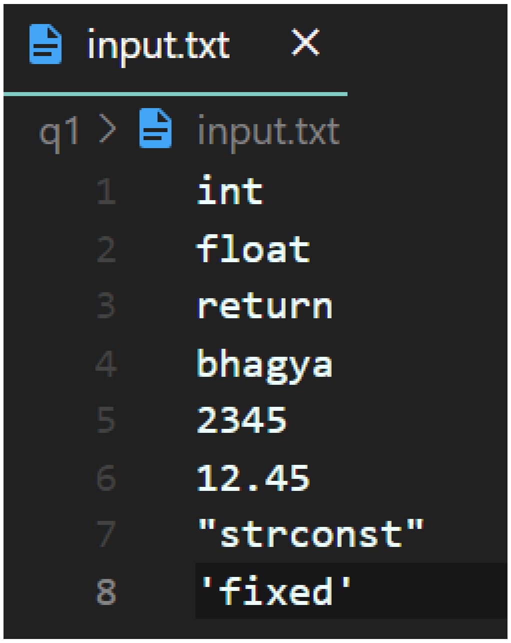
U19CS012

1.) Write a LEX program to Identify **Identifiers**, **Constants** and **Keywords** (*int*, *float*) used in ***C/C++*** from a given Input File.

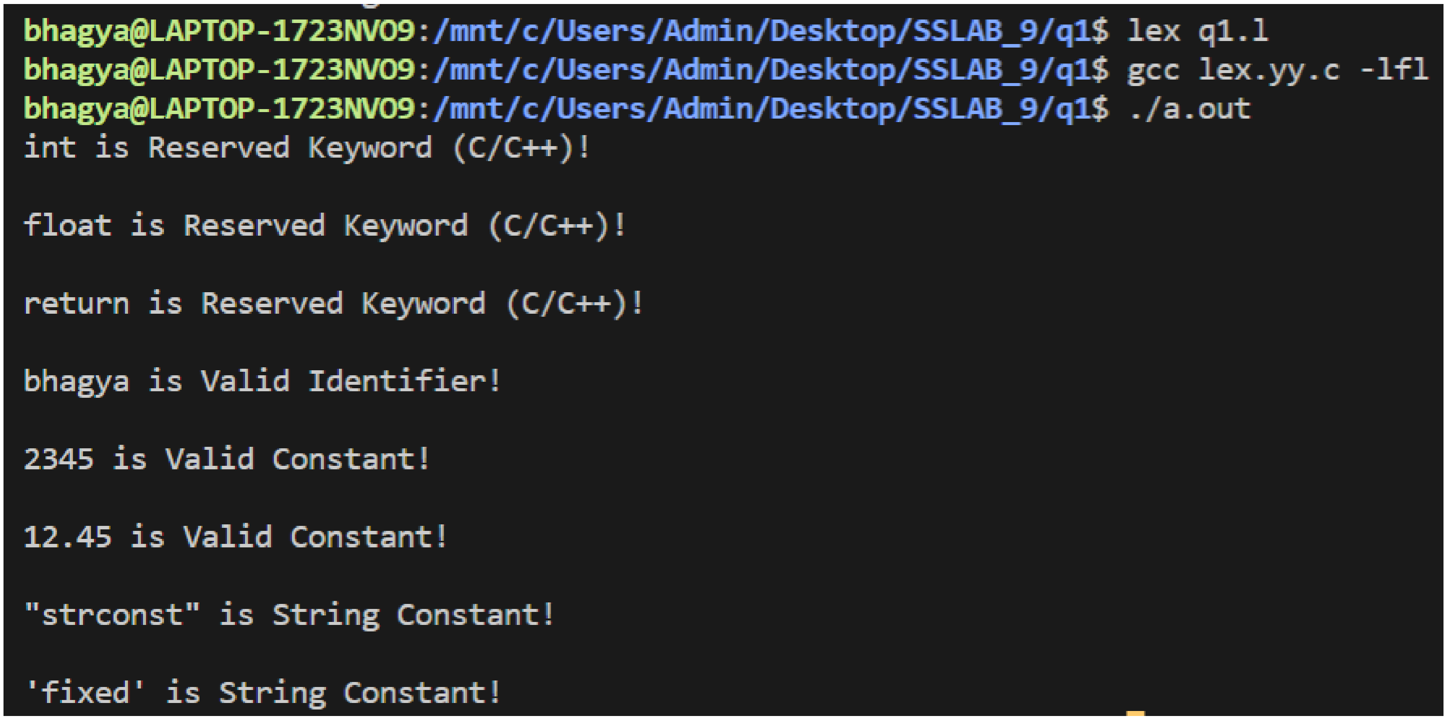
Lex File

```
%{
%}

alphabet [a-zA-Z]
digit [0-9]

%%

asm|double|new|switch|auto|else|operator|template|break|enum|private|this|case|extern|protected|throw|catch|float|public|try|char|for|register|typedef|class|friend|return|union|const|goto|short|unsigned|continue|if|signed|virtual|default|inline|sizeof|void|delete|int|static|volatile|do|long|struct|while { printf("%s is Reserved Keyword (C/C++)!\n",yytext);}

([_]|({alphabet}))([_]|({alphabet})|({digit}))* { printf("%s is Valid Identifier!\n",yytext);}

({digit})+|(({digit})+[.]({digit})+) { printf("%s is Valid Constant!\n",yytext);}

["](.)*["] {printf("%s is String Constant!\n",yytext);}
['']({alphabet})*[''] {printf("%s is String Constant!\n",yytext);}

%%

int main()
{
    yyin=fopen("input.txt","r");
    yylex();
    return 0;
}
```

Output



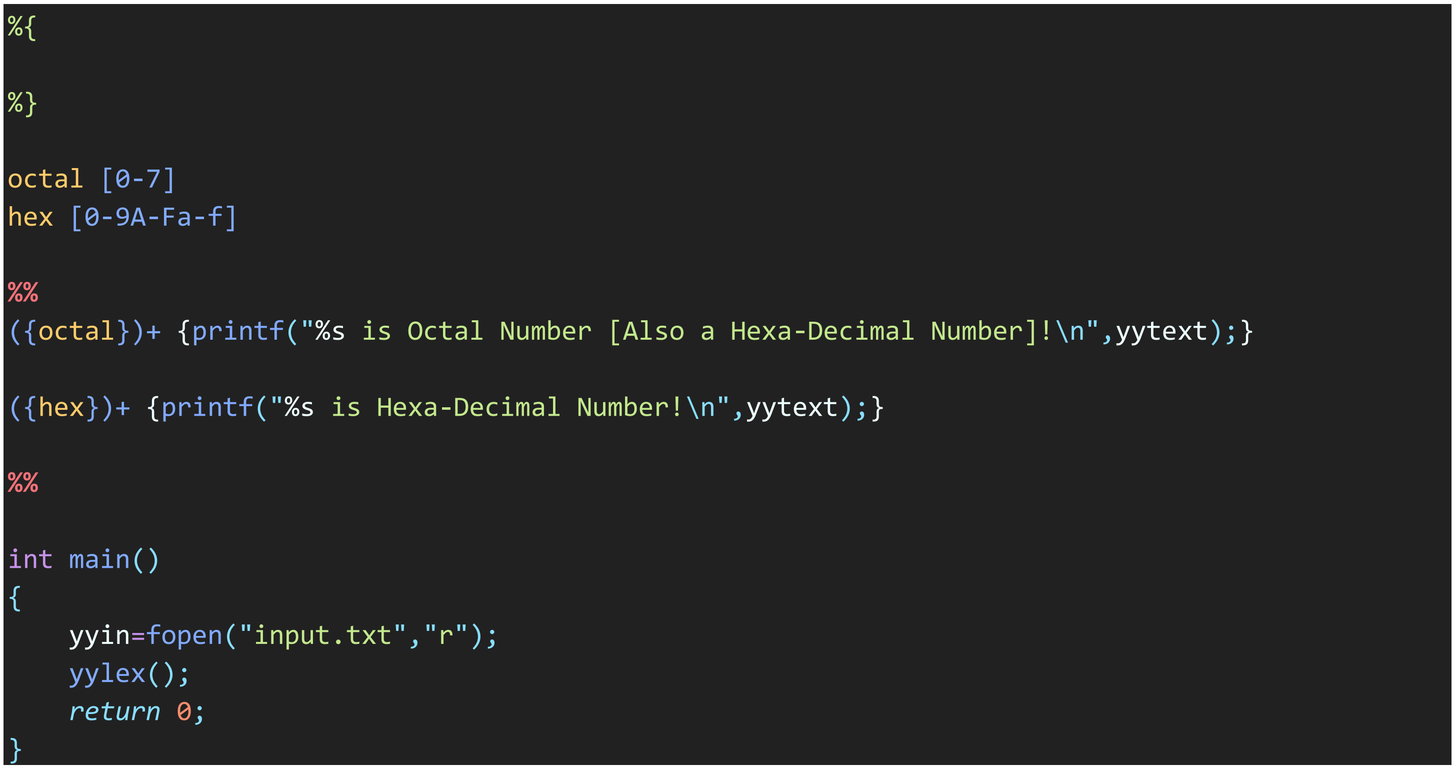
```
int
float
return
bhagya
2345
12.45
"strconst"
'fixed'
```



```
bhagya@LAPTOP-1723NV09:/mnt/c/Users/Admin/Desktop/SSLAB_9/q1$ lex q1.l
bhagya@LAPTOP-1723NV09:/mnt/c/Users/Admin/Desktop/SSLAB_9/q1$ gcc lex.yy.c -lfl
bhagya@LAPTOP-1723NV09:/mnt/c/Users/Admin/Desktop/SSLAB_9/q1$ ./a.out
int is Reserved Keyword (C/C++)!

float is Reserved Keyword (C/C++)!

return is Reserved Keyword (C/C++)!

bhagya is Valid Identifier!

2345 is Valid Constant!

12.45 is Valid Constant!

"strconst" is String Constant!

'fixed' is String Constant!
```

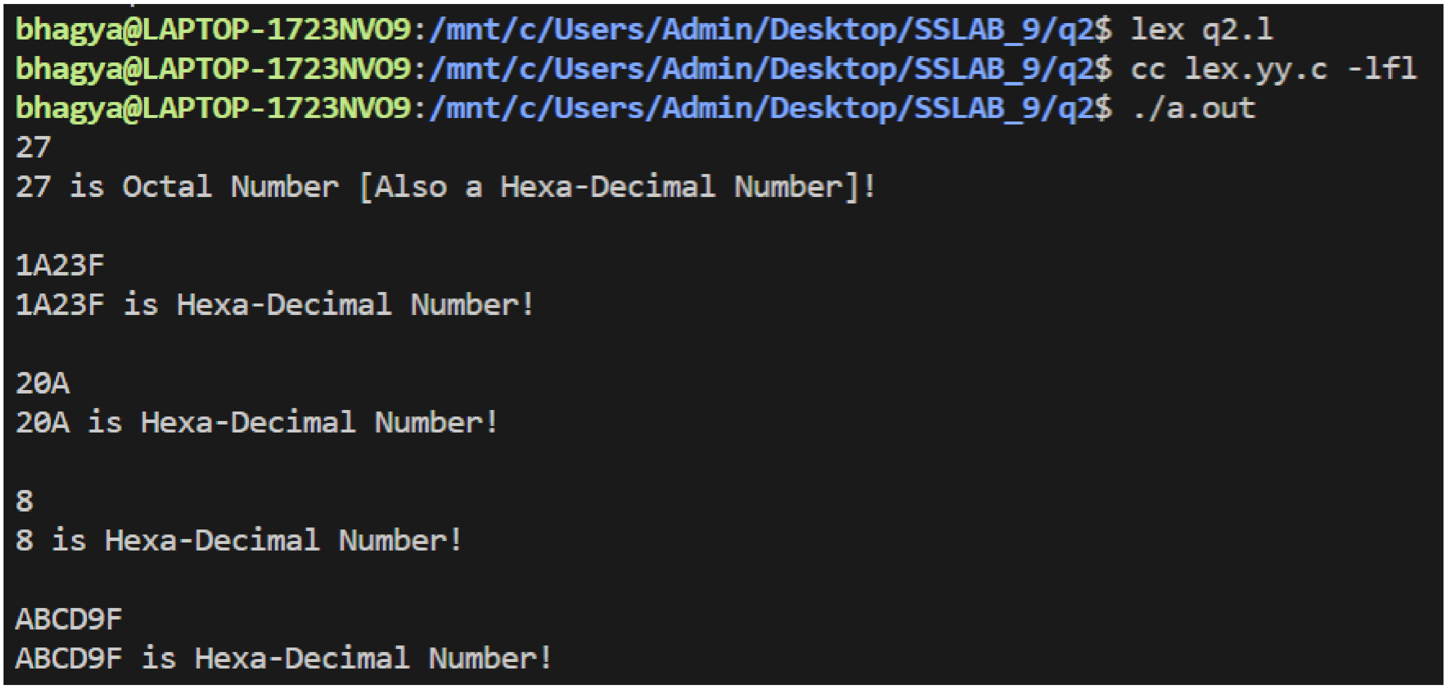
2.) Write a LEX Program to find **Octal** and **Hexadecimal** numbers.

Lex File



```
%{

%}

octal [0-7]
hex [0-9A-Fa-f]

%%
({octal})+ {printf("%s is Octal Number [Also a Hexa-Decimal Number]!\n",yytext);}

({hex})+ {printf("%s is Hexa-Decimal Number!\n",yytext);}

%%

int main()
{
    yyin=fopen("input.txt","r");
    yylex();
    return 0;
}
```

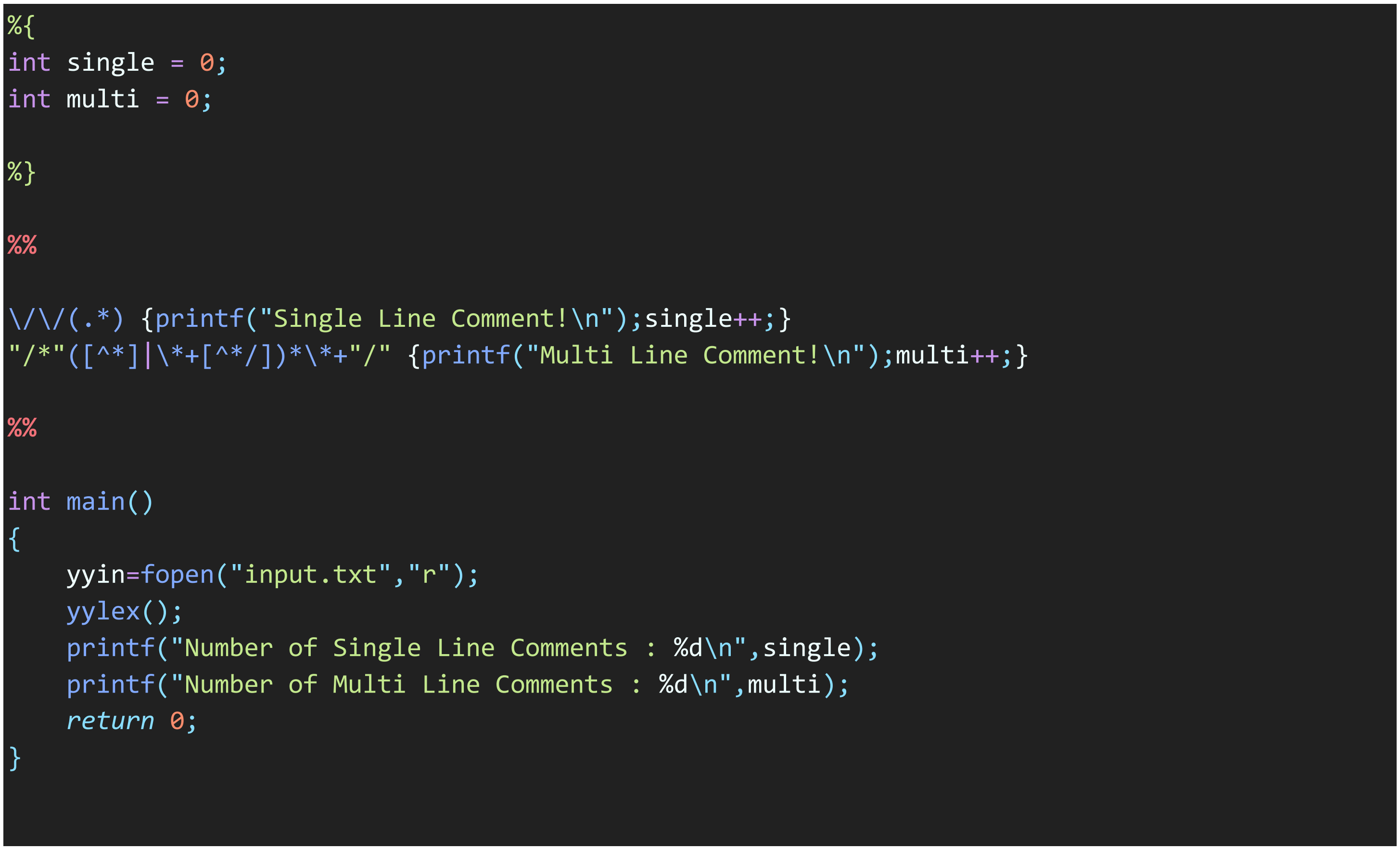
Output



```
bhagya@LAPTOP-1723NV09:/mnt/c/Users/Admin/Desktop/SSLAB_9/q2$ lex q2.l
bhagya@LAPTOP-1723NV09:/mnt/c/Users/Admin/Desktop/SSLAB_9/q2$ cc lex.yy.c -lfl
bhagya@LAPTOP-1723NV09:/mnt/c/Users/Admin/Desktop/SSLAB_9/q2$ ./a.out
27
27 is Octal Number [Also a Hexa-Decimal Number]!

1A23F
1A23F is Hexa-Decimal Number!

20A
20A is Hexa-Decimal Number!

8
8 is Hexa-Decimal Number!

ABCD9F
ABCD9F is Hexa-Decimal Number!
```

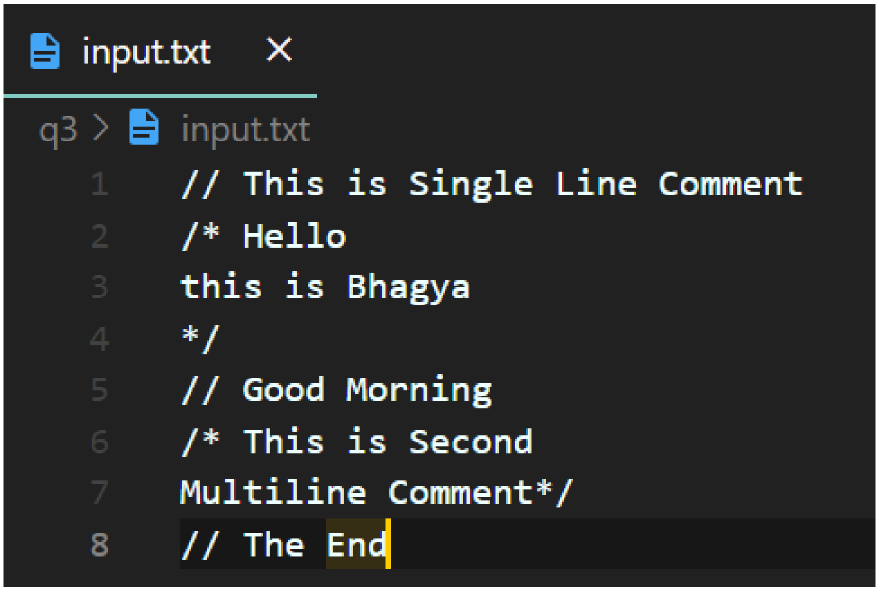
3. Write a LEX Program to **Count** and Display Single line and Multiline comments.

Lex File

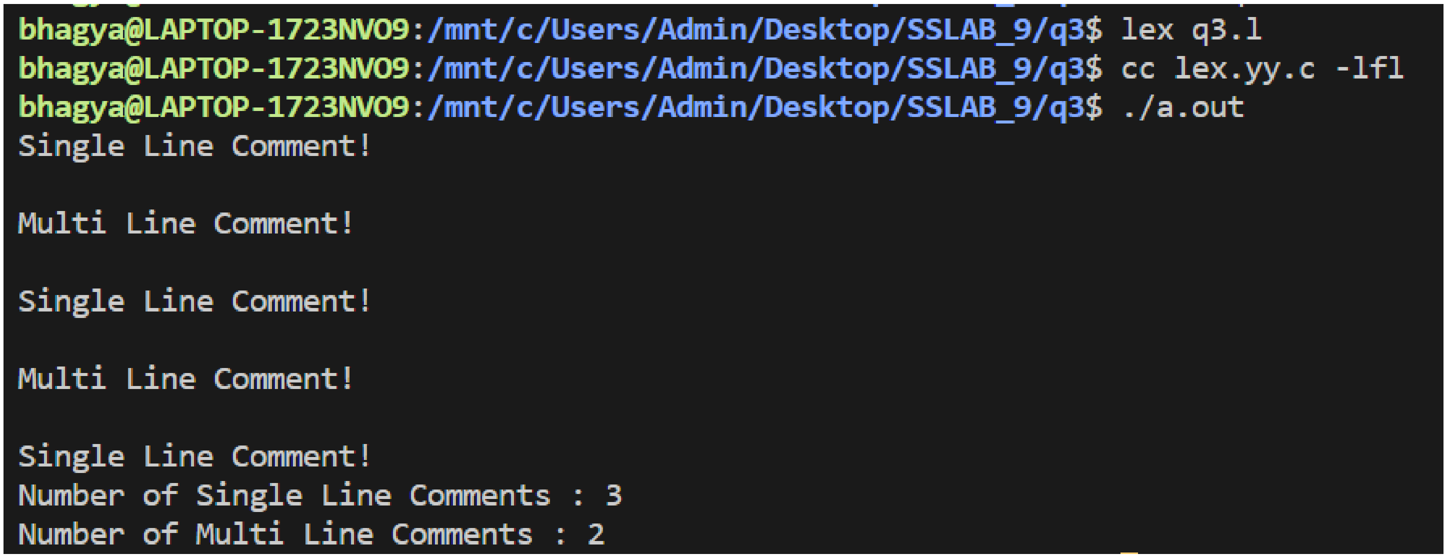


```
%{
int single = 0;
int multi = 0;

%}

%%

\/\/(.*) {printf("Single Line Comment!\n");single++;}
"/*"([^*]|\*+[^*/])*\*+"/" {printf("Multi Line Comment!\n");multi++;}

%%

int main()
{
    yyin=fopen("input.txt","r");
    yylex();
    printf("Number of Single Line Comments : %d\n",single);
    printf("Number of Multi Line Comments : %d\n",multi);
    return 0;
}
```

Output



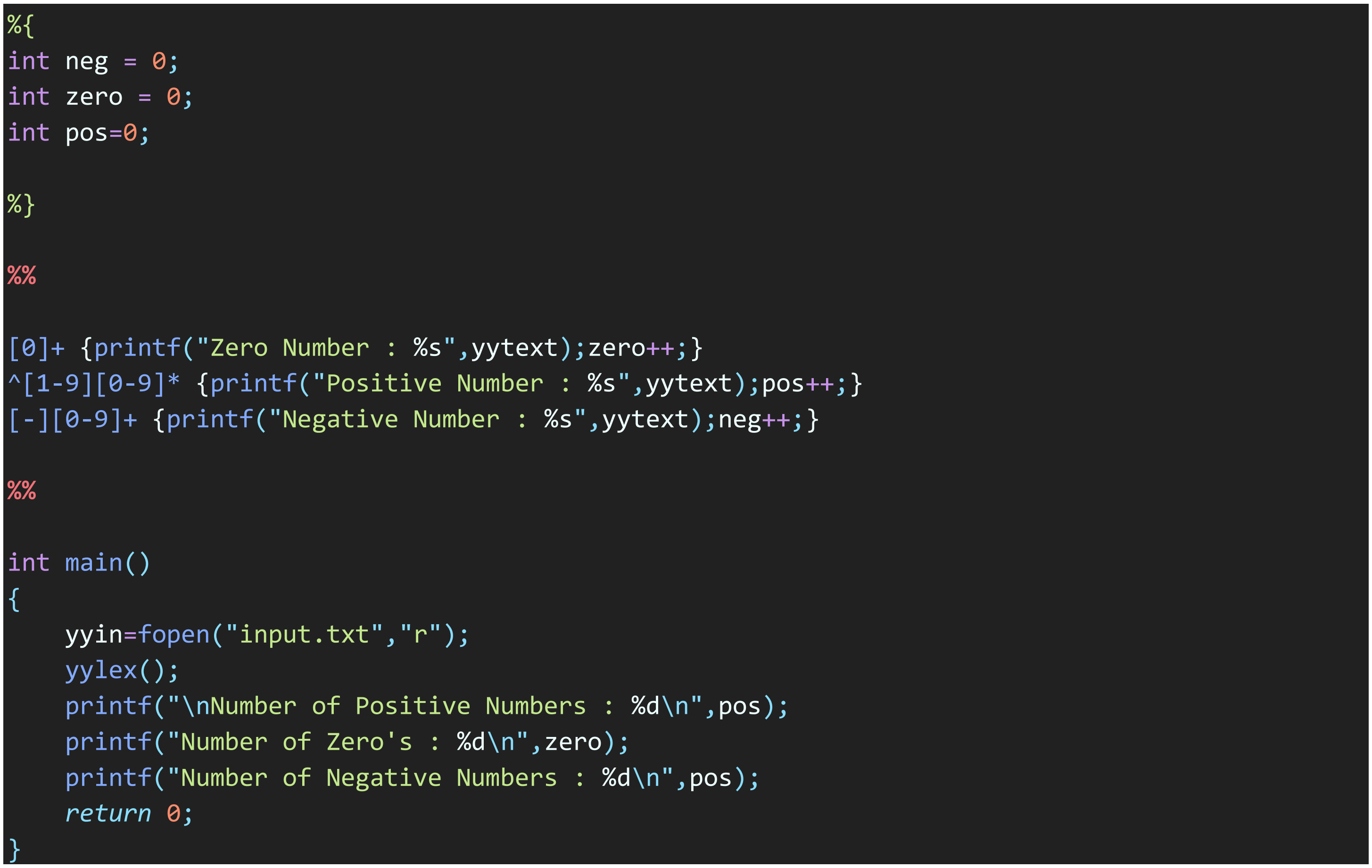
```
// This is Single Line Comment
/* Hello
this is Bhagya
*/
// Good Morning
/* This is Second
Multiline Comment*/
// The End
```



```
bhagya@LAPTOP-1723NVO9:/mnt/c/Users/Admin/Desktop/SSLAB_9/q3$ lex q3.l
bhagya@LAPTOP-1723NVO9:/mnt/c/Users/Admin/Desktop/SSLAB_9/q3$ cc lex.yy.c -lfl
bhagya@LAPTOP-1723NVO9:/mnt/c/Users/Admin/Desktop/SSLAB_9/q3$ ./a.out
Single Line Comment!

Multi Line Comment!

Single Line Comment!

Multi Line Comment!

Single Line Comment!
Number of Single Line Comments : 3
Number of Multi Line Comments : 2
```

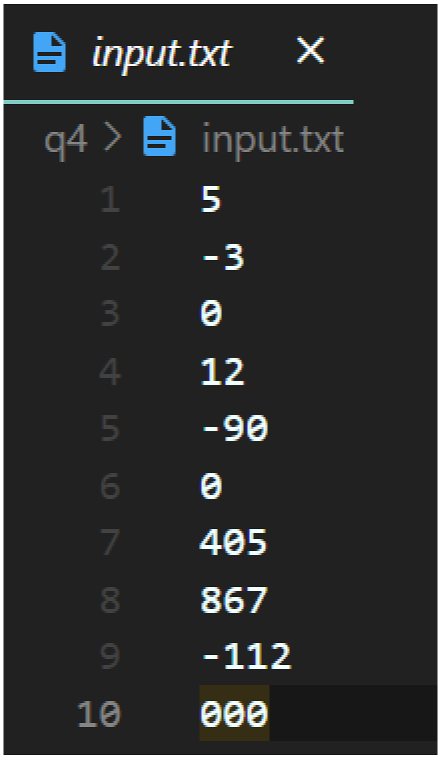
4.) Write a LEX Program to **Count** no of Negative, Positive and Zero numbers.

Lex File

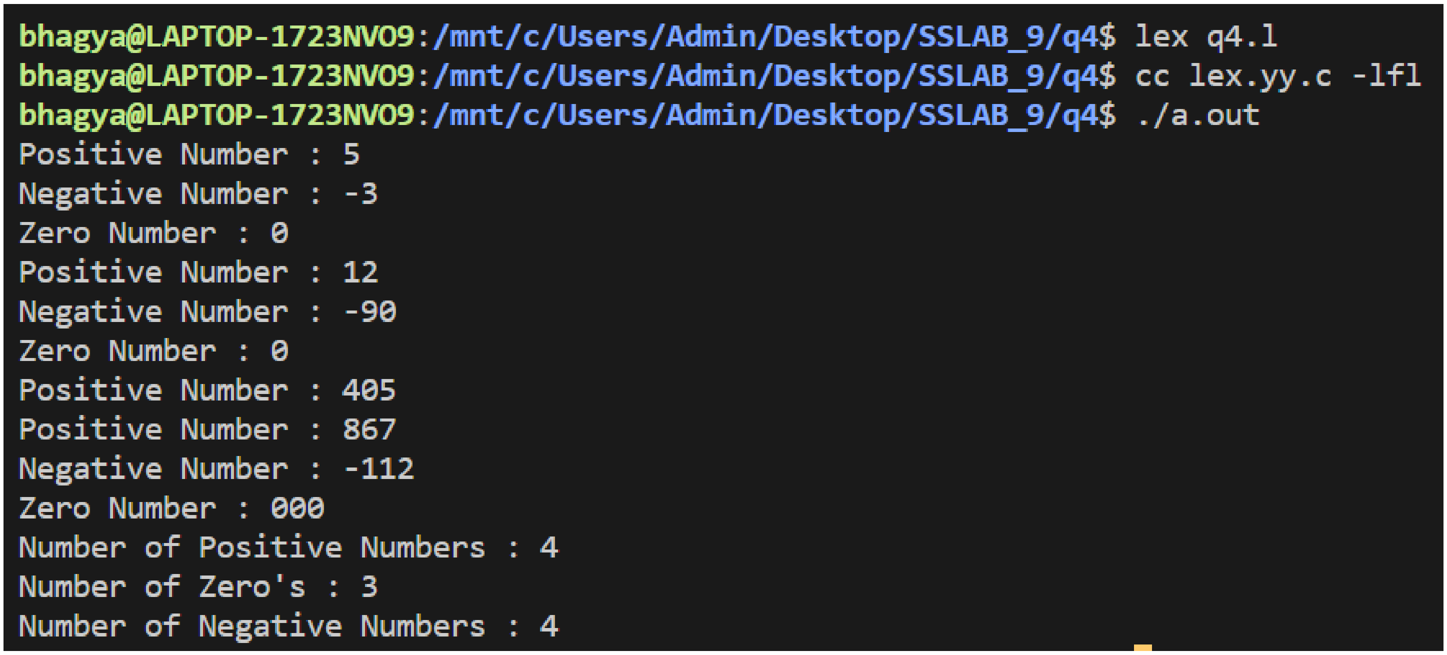


```
%{
int neg = 0;
int zero = 0;
int pos=0;

%}

%%

[0]+ {printf("Zero Number : %s",yytext);zero++;}
^[1-9][0-9]* {printf("Positive Number : %s",yytext);pos++;}
[-][0-9]+ {printf("Negative Number : %s",yytext);neg++;}

%%

int main()
{
    yyin=fopen("input.txt","r");
    yylex();
    printf("\nNumber of Positive Numbers : %d\n",pos);
    printf("Number of Zero's : %d\n",zero);
    printf("Number of Negative Numbers : %d\n",pos);
    return 0;
}
```

Output



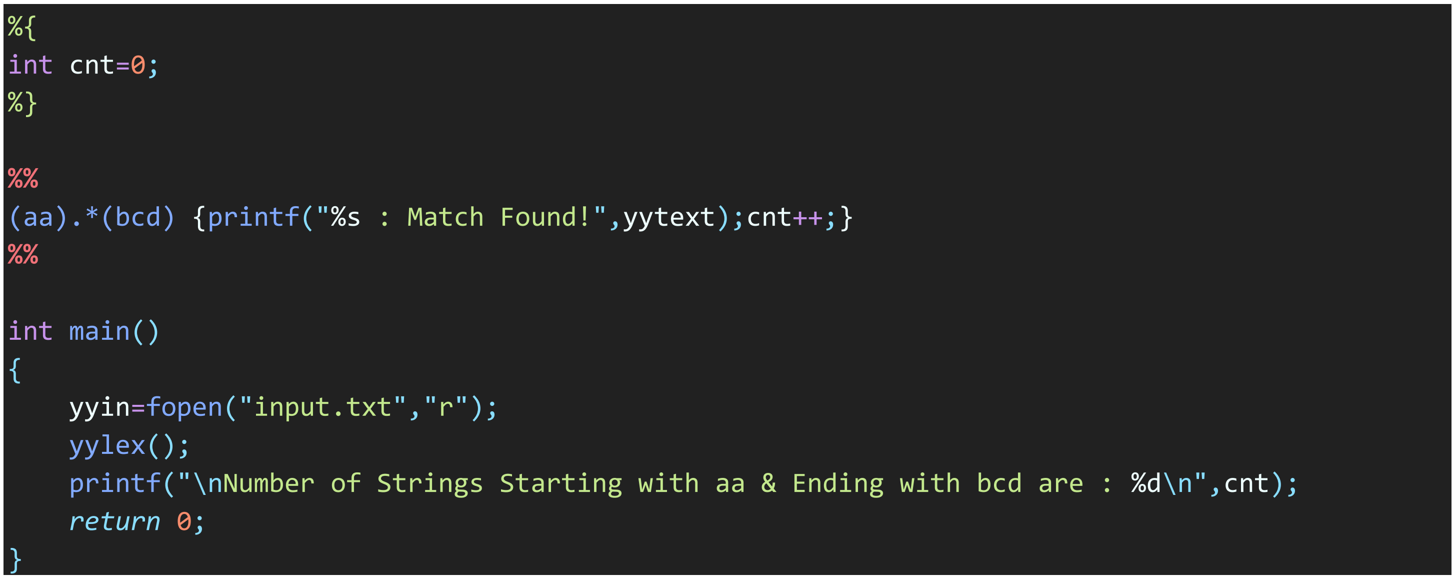
```
5
-3
0
12
-90
0
405
867
-112
000
```



```
bhagya@LAPTOP-1723NVO9:/mnt/c/Users/Admin/Desktop/SSLAB_9/q4$ lex q4.l
bhagya@LAPTOP-1723NVO9:/mnt/c/Users/Admin/Desktop/SSLAB_9/q4$ cc lex.yy.c -lfl
bhagya@LAPTOP-1723NVO9:/mnt/c/Users/Admin/Desktop/SSLAB_9/q4$ ./a.out
Positive Number : 5
Negative Number : -3
Zero Number : 0
Positive Number : 12
Negative Number : -90
Zero Number : 0
Positive Number : 405
Positive Number : 867
Negative Number : -112
Zero Number : 000
Number of Positive Numbers : 4
Number of Zero's : 3
Number of Negative Numbers : 4
```

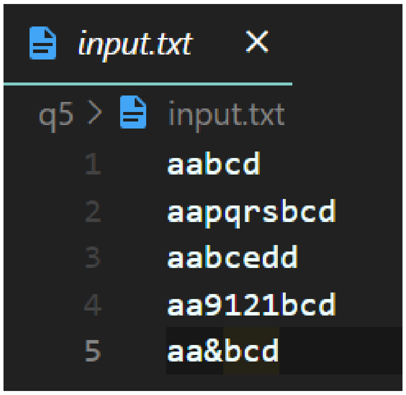
5.) Write a LEX Program to Accept **Strings** that start with **aa** and end with **bcd**.

Lex File

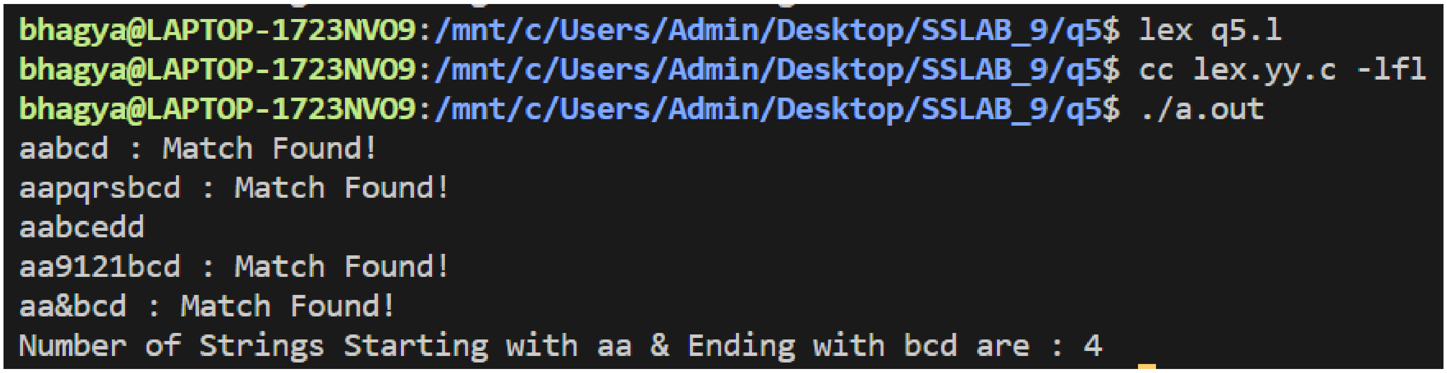


```
%{
int cnt=0;
%}

%%
(aa).*(bcd) {printf("%s : Match Found!",yytext);cnt++;}
%%

int main()
{
    yyin=fopen("input.txt","r");
    yylex();
    printf("\nNumber of Strings Starting with aa & Ending with bcd are : %d\n",cnt);
    return 0;
}
```

Output



```
aabcd
aapqrsbcd
aabcedd
aa9121bcd
aa&bcd
```



```
bhagya@LAPTOP-1723NV09:/mnt/c/Users/Admin/Desktop/SSLAB_9/q5$ lex q5.l
bhagya@LAPTOP-1723NV09:/mnt/c/Users/Admin/Desktop/SSLAB_9/q5$ cc lex.yy.c -lfl
bhagya@LAPTOP-1723NV09:/mnt/c/Users/Admin/Desktop/SSLAB_9/q5$ ./a.out
aabcd : Match Found!
aapqrsbcd : Match Found!
aabcedd
aa9121bcd : Match Found!
aa&bcd : Match Found!
Number of Strings Starting with aa & Ending with bcd are : 4
```

SUBMITTED BY: U19CS012

BHAGYA VINOD RANA