**Exp. No. 21**

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.

**Input Source Program: (sample.c)**

#include <stdio.h>

int main()

{

int number1, number2, sum;

printf("Enter two integers: ");

scanf("%d %d", &number1, &number2);

sum = number1 + number2;

printf("%d + %d = %d", number1, number2, sum);

return 0;

}

**Program: (count\_lines.l)**

%{

int nchar, nword, nline;

%}

%%

\n { nline++; nchar++; }

[^ \t\n]+ { nword++, nchar += yyleng; }

. { nchar++; }

%%

int yywrap(void) {

return 1;

}

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

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

yylex();

printf("Number of characters = %d\n", nchar);

printf("Number of words = %d\n", nword);

printf("Number of lines = %d\n", nline);

fclose(yyin);

}

**Output:**

G:\lex>flex count\_line.l

G:\lex>gcc lex.yy.c

G:\lex>a.exe sample.c

Number of characters = 233

Number of words = 33

Number of lines = 10

G:\lex>

**Exp. No. 22**

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

# **Input Source Program: (sample.c)**

#define P 314

#include<stdio.h> #include<conio.h>

void main()

{

int a,b,c = 30;

printf("hello");

}

**Program: (countconstants.l)**

digit [0-9]

%{

int cons=0;

%}

%%

{digit}+ { cons++; printf("%s is a constant\n", yytext); }

.|\n { }

%%

int yywrap(void) {

return 1; }

int main(void)

{

FILE \*f;

char file[10];

printf("Enter File Name : ");

scanf("%s",file);

f = fopen(file,"r");

yyin = f;

yylex();

printf("Number of Constants : %d\n", cons);

fclose(yyin);

}

**Output:**

G:\lex>flex countconstants.l

G:\lex>gcc lex.yy.c

G:\lex>a.exe

Enter File Name : sample.c

314 is a constant

30 is a constant

Number of Constants : 2

G:\lex>

**Exp. No. 23**

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

# **Input Source Program: (sample.c)**

#define PI 3.14

#include<stdio.h>

#include<conio.h>

void main()

{

int a,b,c = 30;

printf("hello");

}

**Program: (count\_macro.l)**

%{

int nmacro, nheader;

%}

%%

^#define { nmacro++; }

^#include { nheader++; }

.|\n { }

%%

int yywrap(void) {

return 1;

}

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

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

yylex();

printf("Number of macros defined = %d\n", nmacro);

printf("Number of header files included = %d\n", nheader);

fclose(yyin);

}

**Output:**

G:\lex>flex count\_macro.l

G:\lex>gcc lex.yy.c

G:\lex>a.exe sample.c

Number of macros defined = 1

Number of header files included = 2

G:\lex>

**Exp. No. 24**

Write a LEX program to print all HTML tags in the input file.

**Input Source Program: (sample.html)**

<html>

<body>

<h1>My First Heading</h1>

<p>My first paragraph.</p>

</body>

</html>

**Program: (html.l)**

%{

int tags;

%}

%%

"<"[^>]\*> { tags++; printf("%s \n", yytext); }

.|\n { }

%%

int yywrap(void) {

return 1; }

int main(void)

{

FILE \*f;

char file[10];

printf("Enter File Name : ");

scanf("%s",file);

f = fopen(file,"r");

yyin = f;

yylex();

printf("\n Number of html tags: %d",tags);

fclose(yyin);

}

**Output:**

G:\lex>flex html.l

G:\lex>gcc lex.yy.c

G:\lex>a.exe

Enter File Name : sample.html

<html>

<body>

<h1>

</h1>

<p>

</p>

</body>

</html>

Number of html tags: 8

G:\lex>

**Exp. No. 25**

Write a LEX program which adds line numbers to the given C program file and display the same in the standard output.

# **Input Source Program: (sample.c)**

#define PI 3.14

#include<stdio.h>

#include<conio.h>

void main()

{

int a,b,c = 30;

printf("hello");

}

**Program: (addlinenos.l)**

%{

int yylineno;

%}

%%

^(.\*)\n printf("%4d\t%s", ++yylineno, yytext);

%%

int yywrap(void) {

return 1;

}

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

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

yylex();

fclose(yyin);

}

**Output:**

G:\lex>flex addlinenos.l

G:\lex>gcc lex.yy.c

G:\lex>a.exe sample.c

1 #define PI 3.14

2 #include<stdio.h>

3 #include<conio.h>

4 void main()

5 {

6 int a,b,c = 30;

7 printf("hello");

8 }

9

G:\lex>

**Exp. No. 26**

Write a LEX program to count the number of comment lines in a given C program and eliminate them and write into another file.

# **Input Source File: (input.c)**

#include<stdio.h>

int main()

{

int a,b,c; /\*varible declaration\*/ printf(“enter two numbers”); scanf(“%d %d”,&a,&b); c=a+b;//adding two numbers printf(“sum is %d”,c);

return 0;

}

**Program: (comment.l)**

%{

int com=0;

%}

%s COMMENT

%%

"/\*" {BEGIN COMMENT;}

<COMMENT>"\*/" {BEGIN 0; com++;}

<COMMENT>\n {com++;}

<COMMENT>. {;}

\/\/.\* {; com++;}

.|\n {fprintf(yyout,"%s",yytext);}

%%

void main(int argc, char \*argv[])

{

if(argc!=3)

{

printf("usage : a.exe input.c output.c\n");

exit(0);

}

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

yyout=fopen(argv[2],"w");

yylex();

printf("\n number of comments are = %d\n",com);

}

int yywrap()

{

return 1;

}

**Output:**

G:\lex>flex comment.l

G:\lex>gcc lex.yy.c

G:\lex>a.exe input.c

usage : a.exe input.c output.c

G:\lex>a.exe input.c output.c

number of comments are = 2

G:\lex>

# **Output File: (output.c)**

include<stdio.h>

int main()

{

int a,b,c;

printf(“enter two numbers”);

scanf(“%d %d”,&a,&b);

c=a+b;

printf(“sum is %d”,c);

return 0;

}

**Exp. No. 27**

Write a LEX program to identify the capital words from the given input.

**Program: (capital.l)**

%%

[A-Z]+[\t\n ] { printf("%s is a capital word\n",yytext); }

. ;

%%

int main( )

{

printf("Enter String :\n");

yylex();

}

int yywrap( )

{

return 1;

}

**Output:**

G:\lex>flex capital.l

G:\lex>gcc lex.yy.c

G:\lex>a.exe

Enter String :

CAPITAL of INDIA is DELHI

CAPITAL is a capital word

INDIA is a capital word

DELHI

is a capital word

G:\lex>

**Exp. No. 28**

Write a LEX Program to check the email address is valid or not.

**Program: (**email\_valid.l)

%{

int flag=0;

%}

%%

[a-z . 0-9]+@[a-z]+".com"|".in" { flag=1; }

%%

int main()

{

yylex();

if(flag==1)

printf("Accepted");

else

printf("Not Accepted");

}

int yywrap()

{ return 1;

}

**Output:**

G:\lex>flex email\_valid.l

G:\lex>gcc lex.yy.c

G:\lex>a.exe

sse123@gmail.com

Accepted

G:\lex>

**Exp. No. 29**

Write a LEX Program to convert the substring abc to ABC from the given input string

**Program: (substring.l)**

%{

int i;

%}

%%

[a-z A-Z]\* { for(i=0;i<=yyleng;i++)

{ if((yytext[i]=='a')&&(yytext[i+1]=='b')&&(yytext[i+2]=='c'))

{ yytext[i]='A';

yytext[i+1]='B';

yytext[i+2]='C';

}

}

printf("%s",yytext);

}

[\t]\* return 1;

.\* {ECHO;}

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

%%

int main()

{

yylex();

}

int yywrap()

{

return 1;

}

**Output:**

G:\lex>flex substring.l

G:\lex>gcc lex.yy.c

G:\lex>a.exe

abcdefghabcijkla

ABCdefghABCijkla

G:\lex>

**Exp. No. 30**

Implement a LEX program to check whether the mobile number is valid or not.

**Program: (mobile.l)**

%%

[1-9][0-9]{9} {printf("\nMobile Number Valid\n");}

.+ {printf("\nMobile Number Invalid\n");}

%%

int main()

{

printf("\nEnter Mobile Number : ");

yylex();

printf("\n");

return 0;

}

int yywrap()

{ }

**Output:**

G:\lex>flex mobile.l

G:\lex>gcc lex.yy.c

G:\lex>a.exe

Enter Mobile Number : 7856453489

Mobile Number Valid

G:\lex>