EXECUTION – EXCERSISES

IMPLEMENTATION OF A LEXICAL ANALYSER USING LEX

1.Write a lex program that replaces the substring abc by ABC from the given input string.

CODE :-

```
%option noyywrap
%{
        #include<stdio.h>
        int flag=0;
%}

%%
(abc) {
        printf("ABC");
        flag++;
}
. {
        printf("%s", yytext);
        }
%%
int main()
{
        yylex();
        if(flag>=1){
        printf("\n \'abc\' has been changed in %d instances.\n", flag);
        }else{
        printf("\n No substring matches \'abc\'.\n");
        }
        return 0;
}
```

OUTPUT :-

```
D:\STUDIES\SEM 5\CD\LAB\CODE\LAB 4>lex p1.l

D:\STUDIES\SEM 5\CD\LAB\CODE\LAB 4>gcc lex.yy.c

D:\STUDIES\SEM 5\CD\LAB\CODE\LAB 4>a.exe
abc is a substring of abcde
ABC is a substring of ABCde


Whereas ABC is not a substring of abcde
Whereas ABC is not a substring of ABCde


 'abc' has been changed in 3 instances.
```

```
D:\STUDIES\SEM 5\CD\LAB\CODE\LAB 4>a.exe
Hello World abc DEF ghi
Hello World ABC DEF ghi


Cool this converts abc to ABC!!!
Cool this converts ABC to ABC!!!


abc ghiabc mi cskabc rcb kkrabcpbks rr srh
ABC ghiABC mi cskABC rcb kkrABCpbks rr srh


 'abc' has been changed in 6 instances.
```

2.Write a lex program to identify whether a given Well formedness of bracket.

CODE :-

```
%option noyywrap
%{
        #include<stdio.h>
        int left=0, right=0, line=1;
%}

%%
"(" {left++;}
")" {right++;}

[\n] {
        if(left==right){
                printf("Yes. The given line has no missing parantesis.\n", line);
        }else if(left<right){
                printf("No. The given line has either extra ) paranthesis or missing ( parantesis.\n", line);
        }else if(left>right) {
                printf("No. The given line has either extra ( paranthesis or missing ) parantesis.\n", line);
        }

        left=0;
        right=0;
        line++;
}
%%
int main()
{
        yylex();
        return 0;
}
```

OUTPUT :-

```
D:\STUDIES\SEM 5\CD\LAB\CODE\LAB 4>lex p2.l

D:\STUDIES\SEM 5\CD\LAB\CODE\LAB 4>gcc lex.yy.c

D:\STUDIES\SEM 5\CD\LAB\CODE\LAB 4>a.exe
(()(()))
Yes. The given line has no missing parantesis.
((()))(((()
No. The given line has either extra ( paranthesis or missing ) parantesis.
(()()()()))
No. The given line has either extra ) paranthesis or missing ( parantesis.
(()()()()()()
Yes. The given line has no missing parantesis.
```

3.Write a lex program to finding vowels and consonant in a string.

CODE :-

```
%option noyywrap
%{
        #include<stdio.h>
        int vow=0,cons=0;
%}

%%
[aeiouAEIOU] {vow++;}
[a-zA-Z] {cons++;}
[\n] {
        printf("There are %d vowels and %d consonants.\n", vow, cons);
        vow=0;
        cons=0;
        }
%%
int main()
{
        yylex();
        return 0;
}
```

OUTPUT :-

```
D:\STUDIES\SEM 5\CD\LAB\CODE\LAB 4>lex p3.l

D:\STUDIES\SEM 5\CD\LAB\CODE\LAB 4>gcc lex.yy.c

D:\STUDIES\SEM 5\CD\LAB\CODE\LAB 4>a.exe
Hello World
 There are 3 vowels and 7 consonants.
Sachin Raghul
 There are 4 vowels and 8 consonants.
Compiler Design
 There are 5 vowels and 9 consonants.
```

4.Write a lex program to finding the capital letter in given input string.

CODE :-

```
%option noyywrap
%{
        #include<stdio.h>
%}

%%
[A-Z] { printf("%s, ",yytext); }
. ;
%%
int main()
{
        yylex();
        printf("\nThese are the capital letters in the given input\n");
        return 0;
}
```

OUPUT :-

```
D:\STUDIES\SEM 5\CD\LAB\CODE\LAB 4>lex p4.l

D:\STUDIES\SEM 5\CD\LAB\CODE\LAB 4>gcc lex.yy.c

D:\STUDIES\SEM 5\CD\LAB\CODE\LAB 4>a.exe
Computer Science and Engineering
C, S, E,
Compiler Design
C, D,
Indian Administrative Service
I, A, S,
College of Engineering, Guindy
C, E, G,
Indian Premier League
I, P, L,


These are the capital letters in the given input
```