Lex program to count the frequency of the given word in a file

lex countWord.l
gcc lex.yy.c
./a.out

```
%{
#include<stdio.h>
#include<string.h>

char word [] = "geeks";
int count = 0;


%}

/* Rule Section */
/* Rule 1 compares the matched token with the
   word to count and increments the count variable
   on successful match  */
/* Rule 2 matches everything other than string
   (consists of alphabets only ) and do nothing */

%%
[a-zA-Z]+  { if(strcmp(yytext, word)==0)
                    count++; }
.  ;
%%


int yywrap()
{
    return 1;
}

/* code section */
int main()
{
        extern FILE *yyin, *yyout;

        /* open the input file
           in read mode */
        yyin=fopen("input.txt", "r");
        yylex();
```

```
        printf("%d", count);

}
```

Lex Program to remove comments from C program

lex abc.l (abc is the file name)
cc lex.yy.c -efl
./a.out

```
/% Lex Program to remove comments from C program
   and save it in a file %/
/*Definition Section*/
%{
%}

/*Starting character sequence for multiline comment*/
start \/\*
/*Ending character sequence for multiline comment*/
end   \*\/

/*Rule Section*/
%%

/*Regular expression for single line comment*/
\/\/(.*) ;
/*Regular expression for multi line comment*/
{start}.*{end} ;

%%

/*Driver function*/
int main(int k,char **argcv)
{
yyin=fopen(argcv[1],"r");
yyout=fopen("out.c","w");
/*call the yylex function.*/
yylex();
return 0;
```

```
}
```

Lex code to count total number of tokens

Lex 4.l

Cc lex.yy.c - lfl

./a.out

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

// rule section
%%

//count number of keywords
"while"|"if"|"else" {n++;printf("\t keywords : %s", yytext);}

// count number of keywords
"int"|"float" {n++;printf("\t keywords : %s", yytext);}

// count number of identifiers
[a-zA-Z_][a-zA-Z0-9_]* {n++;printf("\t identifier : %s", yytext);}

// count number of operators
"<="|"=="|"="|"++"|"-"|"*"|"+" {n++;printf("\t operator : %s",
yytext);}

// count number of separators
[(){}|, ;] {n++;printf("\t separator : %s", yytext);}

// count number of floats
[0-9]*"."[0-9]+ {n++;printf("\t float : %s", yytext);}
```

```
// count number of integers
[0-9]+ {n++;printf("\t integer : %s", yytext);}

.     ;
%%



int main()

{

    yylex();

    printf("\n total no. of token = %d\n", n);

}
```

Check Even Or odd

Lex evenOrOdd.l
gcc lex.yy.c
./a.out

```
/*Lex program to take check whether
the given number is even or odd */

%{
#include<stdio.h>
int i;
%}

%%

[0-9]+      {i=atoi(yytext);
           if(i%2==0)
                printf("Even");
           else
          printf("Odd");}
```

```
%%

int yywrap(){}

/* Driver code */
int main()
{

    yylex();
    return 0;
}
```