Name: Brijesh Rameshbhai Rohit
Admission number : U19CS009

SS-ASSIGNMENT-08

1. Write a Lex program to count the number of lines, characters and words of the given input file.

CODE :

```
%{
    #include<stdio.h>
    int countLine = 0, countWords = 0, countChars = 0;
%}


%%
\n {countLine++;}
[^ \n\t]+ {countWords++, countChars+=yyleng-1;}
. {countChars++;}
%%

int yywrap(void){}

int main(int argc, char*argv[])
{
    yyin=fopen(argv[1],"r");
    yylex();

    printf("Number of lines : %d\n", countLine);
    printf("Number of words : %d\n", countWords);
    printf("Number of chars : %d\n", countChars);
}
```

INPUT TEXT FILE :

```
input.txt  ×    count.l

count > input.txt
   1    Hello there!!!
   2    I'm Brijesh Rohit
   3    How are you doing?
   4    I'm good.
   5    Have a nice day.
   6    Bye!!!!
   7    Take care.
```

OUTPUT :

```
(base) brijesh@pop-os-birju:~/Documents/ss/ss-assign08/count$ ./a.out input.txt
Number of lines : 6
Number of words : 18
Number of chars : 73
```

2. Write a lex program to find out the total number of vowels, and consonants from the given input string.

CODE :

```
%{
   #include<stdio.h>
   int vowels=0, consonants=0, chars=0;
%}

%%
[aeiouAEIOU] {vowels++;chars++;}
[a-zA-Z] {consonants++;chars++;}
. {chars++;}
%%
```

```
int yywrap() {}

int main() {
    printf("Enter string : ");
    yylex();
    printf("\nNumber of vowel : %d\n", vowels);
    printf("Number of consonants : %d\n", consonants);
    printf("Number of chars : %d\n", chars);
    return 0;
}
```

OUTPUT :

```
(base) brijesh@pop-os-birju:~/Documents/ss/ss-assign08/vowels$ lex vowels.l
(base) brijesh@pop-os-birju:~/Documents/ss/ss-assign08/vowels$ gcc lex.yy.c -lfl
(base) brijesh@pop-os-birju:~/Documents/ss/ss-assign08/vowels$ ./a.out
Enter string : This is a multiline string with more than 20 vowels and other characters like

######@@@!@!$!#$#%@$ are also present and are not necessarily consonants.
Number of vowel : 39
Number of consonants : 67
Number of chars : 150
```

3. Write a Lex Program to convert Lowercase string to Upper case.
Input:abc Output: ABC

CODE :

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

%}

%%
[a-z] printf("%c",yytext[0] - ('a' - 'A'));
[A-Z] printf("%c",yytext[0] + ('a' - 'A'));
. printf("%c",yytext[0]);
%%
```

```
int yywrap(){}

int main(){
    printf("Enter string : ");
    yylex();
    printf("\n");
    return 0;
}
```

OUTPUT :

```
(base) brijesh@pop-os-birju:~/Documents/ss/ss-assign08/changecase$ ./a.out
Enter string : Hello I'm Brijesh Rohit, I'm 21 and my weight os above 200 pounds.
hELLO i'M bRIJESH rOHIT, i'M 21 AND MY WEIGHT OS ABOVE 200 POUNDS.
@!#!#!@#$!#!@#Hello there, Will you be my Date????
@!#!#!@#$!#!@#hELLO THERE, wILL YOU BE MY dATE????
^C
(base) brijesh@pop-os-birju:~/Documents/ss/ss-assign08/changecase$ █
```

4. Write a Lex program to check valid/invalid
(a) Mobile number (considering 10-digit mobile number followed by country code +91)
(b) Email address

CODE :

```
%{
    #include<stdio.h>
    int flag = 0;
%}
%%
[a-z.0-9_]+@[a-z]+".com"|".in"|".org" flag = 1;
[+91][1-9][0-9]{9} {if(yyleng == 13)flag = 1;}
%%
int yywrap(){}
int main()
{
```

```
    printf("Please enter your email/mobile number with initials :
");
    yylex();
    printf("\n");
    int flag = 1;
    if (flagemail == 1)
    {
        printf("Accepted\n");
        flag = 0;
    }
    if(flag)
        printf("Invalid input\n");
    return 0;
}
```

OUTPUT :

```
(base) brijesh@pop-os-birju:~/Documents/ss/ss-assign08/validate$ ./a.out
Please enter your email/mobile number with initials : brijeshrohit1.7@gmail.com
Accepted
(base) brijesh@pop-os-birju:~/Documents/ss/ss-assign08/validate$ ./a.out
Please enter your email/mobile number with initials : brijesh$#@gmail.combrijesh
Invalid input
```

```
(base) brijesh@pop-os-birju:~/Documents/ss/ss-assign08/validate$ ./a.out
Please enter your email/mobile number with initials : +919978102374
Accepted
(base) brijesh@pop-os-birju:~/Documents/ss/ss-assign08/validate$ ./a.out
Please enter your email/mobile number with initials : +91 9978102374
Invalid input
(base) brijesh@pop-os-birju:~/Documents/ss/ss-assign08/validate$ ./a.out
Please enter your email/mobile number with initials : +91 995335435432456234
Invalid input
```

5. Write a Lex program to implement a simple Calculator.

CODE :

```lex
%{
    #include<stdio.h>
    char op;
    float num1=0, num2=0;
    int flag = 0, flag1 = 0;
%}
%%
([0-9])+ {if(flag==0)
{num1=atoi(yytext);}else{num2=atoi(yytext);}};
([0-9])*[.]([0-9])* {if(flag==0)
{num1=atof(yytext);}else{num2=atof(yytext);}};
([/*+-]) {op=yytext[0];flag=1;};
\n {flag=0;if(op=='+')printf("Answer: %f\n",num1+num2); else
if(op=='-')printf("Answer: %f\n",num1-num2); else
if(op=='*')printf("Answer: %f\n",num1*num2);else if(num2
==0)printf("Division with 0 is invalid\n"); else
if(op=='/')printf("Answer: %f\n",num1/num2);};
. {};
%%
int main()
{
    yylex();
    return 0;
}
```

OUTPUT :

```
(base) brijesh@pop-os-birju:~/Documents/ss/ss-assign08/calculator$ lex cal.l
(base) brijesh@pop-os-birju:~/Documents/ss/ss-assign08/calculator$ gcc lex.yy.c -lfl
(base) brijesh@pop-os-birju:~/Documents/ss/ss-assign08/calculator$ ./a.out
5+5
Answer: 10.000000
1/0
Division with 0 is invalid
1/3
Answer: 0.333333
2.3424323*123123123
Answer: 288407584.000000
1231/1231
Answer: 1.000000
1231-1243123.1231231231
Answer: -1241892.125000
```