NAME:AYANIKA PAUL
Roll No. 22
D1

LAB 5

Q1)Count the number of vowels and consonants in the given input.

CODE:

q1.l

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

%%
[aeiouAEIOU]  { vowels++; }
[b-df-hj-np-tv-zB-DF-HJ-NP-TV-Z] { consonants++; }

.|\n           {}  // Ignore other characters (e.g., spaces, punctuation, etc.)
%%

int main() {
    printf("Enter a string: ");
    yylex();
    printf("Vowels: %d\n", vowels);
    printf("Consonants: %d\n", consonants);
    return 0;
}
int yywrap() {
    return 1;
}
```

OUTPUT:

```
}student@oslab-02:~/220905128/lab5$ flex q1.l
student@oslab-02:~/220905128/lab5$ gcc lex.yy.c -o q1
student@oslab-02:~/220905128/lab5$ ./q1
Enter a string: aeiOU Hello
Vowels: 7
Consonants: 3
student@oslab-02:~/220905128/lab5$ ./q1
Enter a string: Apple
Vowels: 2
Consonants: 3
```

Q2)Count the number of words, characters, blanks and lines in a given text.

CODE:

q1.l | q2.l | q4.l

```
%{
#include <stdio.h>
int words = 0;
int characters = 0;
int blanks = 0;
int lines = 0;
%}

%%
[ \t]            { blanks++; characters++; }
[a-zA-Z0-9]+     { words++; characters += yyleng; }
\n               { lines++; characters++; }
.                { characters++; }
%%

int main() {
    printf("Enter text (Ctrl+D to end input):\n");
    yylex();
    printf("\nWords: %d\n", words);
    printf("Characters: %d\n", characters);
    printf("Blanks (spaces/tabs): %d\n", blanks);
    printf("Lines: %d\n", lines);
    return 0;
}

int yywrap() {
    return 1;
}
```

OUTPUT:

```
student@oslab-02:~/220905128/lab5$ flex q2.l
student@oslab-02:~/220905128/lab5$ gcc lex.yy.c -o q2
student@oslab-02:~/220905128/lab5$ ./q2
Enter text (Ctrl+D to end input):
Hello
This is Lab5
CD lab

Words: 6
Characters: 26
Blanks (spaces/tabs): 3
Lines: 3
```

Q3)Find the number of positive integer, negative integer, positive floating positive number and negative floating point number

CODE:

q3.l | q1.l | q2.l | q4.l

```
%{
#include <stdio.h>
int positive_integers = 0;
int negative_integers = 0;
int positive_floats = 0;
int negative_floats = 0;
%}

%%
[0-9]+               { positive_integers++; }
-[0-9]+              { negative_integers++; }
[0-9]+\.[0-9]+       { positive_floats++; }
-[0-9]+\.[0-9]+      { negative_floats++; }

.|\n                  {} // Ignore other characters (spaces, punctuation, etc.)
%%

int main() {
    printf("Enter the input (Ctrl+D to end):\n");
    yylex();
    printf("\nPositive integers: %d\n", positive_integers);
    printf("Negative integers: %d\n", negative_integers);
    printf("Positive floating point numbers: %d\n", positive_floats);
    printf("Negative floating point numbers: %d\n", negative_floats);
    return 0;
}

int yywrap() {
    return 1;
}
```

OUTPUT:

```
student@oslab-02:~/220905128/lab5$ flex q3.l
student@oslab-02:~/220905128/lab5$ gcc lex.yy.c -o q3
student@oslab-02:~/220905128/lab5$ ./q3
Enter the input (Ctrl+D to end):
123 -123 34.00 2 -99.05 -44

Positive integers: 2
Negative integers: 2
Positive floating point numbers: 1
Negative floating point numbers: 1
```

Q4)Given a input C file, replace all scanf with READ and printf with WRITE statements also find the number of scanf and printf in the file.

CODE:

q4.l

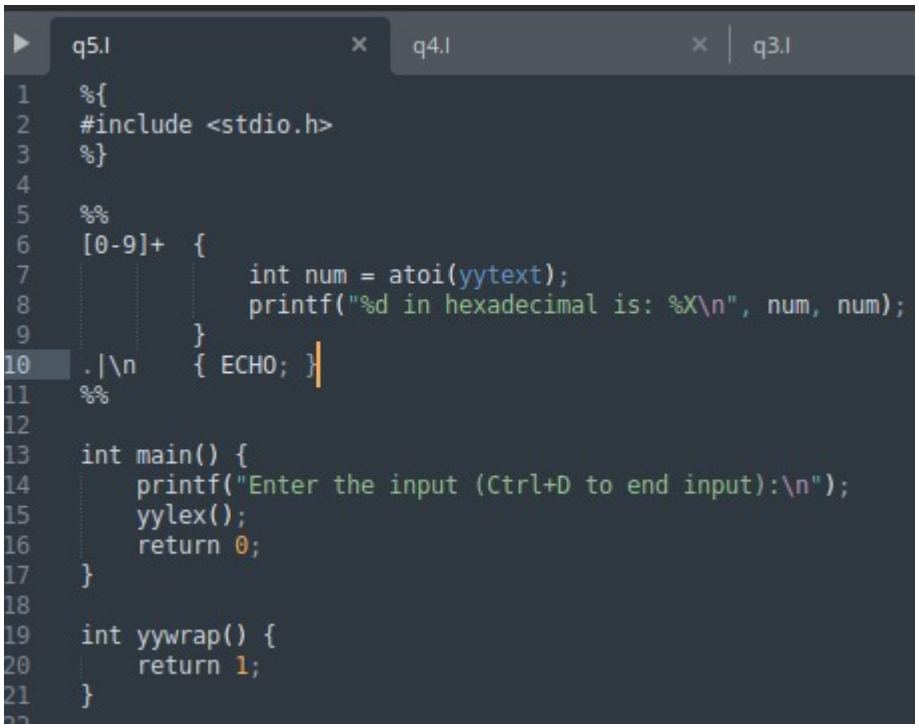
```
%{
#include <stdio.h>
int scanf_count = 0;
int printf_count = 0;
%}

%%
scanf    { printf("READ"); scanf_count++; }
printf         { printf("WRITE"); printf_count++; }
.|\n             { ECHO; }

%%

int main() {
    printf("Enter the C code (Ctrl+D to end input):\n");
    yylex();
    printf("\nNumber of scanf: %d\n", scanf_count);
    printf("Number of printf: %d\n", printf_count);
    return 0;
}
int yywrap() {
    return 1;
}
```

OUTPUT:

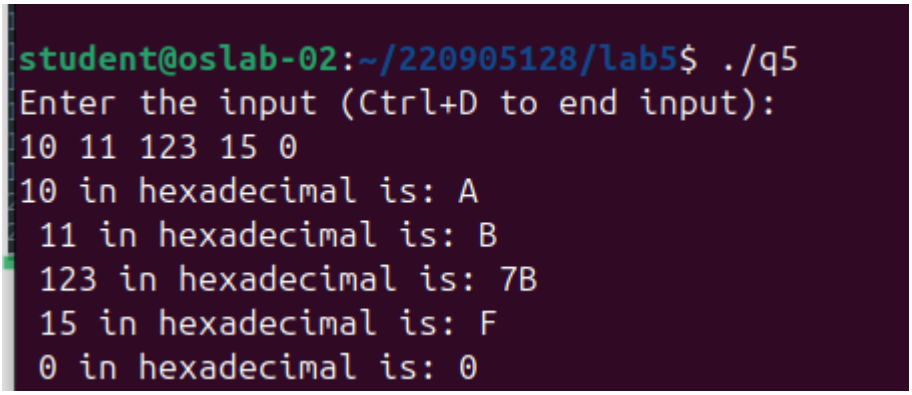
```
student@oslab-02:~/220905128/lab5$ flex q4.l
student@oslab-02:~/220905128/lab5$ gcc lex.yy.c -o q4
student@oslab-02:~/220905128/lab5$ ./q4
Enter the C code (Ctrl+D to end input):
#include<stdio.h>

int main() {
    int num;
    scanf("%d", &num);
    printf("You entered: %d\n", num);
    printf("\n");
    return 0;
#include<stdio.h>

int main() {
    int num;
    READ("%d", &num);
    WRITE("You entered: %d\n", num);
    WRITE("\n");
    return 0;

Number of scanf: 1
Number of printf: 2
```

Q5)That changes a number from decimal to hexadecimal notation.

CODE:



OUTPUT:



Q6)Convert uppercase characters to lowercase characters of C file excluding the characters present in the comment.

CODE:

q6.l | q5.l | q4.l

```
%{
#include <stdio.h>
#include <ctype.h>  // For the tolower() function
%}

%%
"//".*                  { ECHO; }
"/*"[^*]*"*"+[^/]*"/"   { ECHO; }
[a-zA-Z] {
    if (isupper(yytext[0])) {
        yytext[0] = tolower(yytext[0]);
    }
    ECHO;
}
.|\n { ECHO; }
%%

int main() {
    printf("Enter the C code (Ctrl+D to end input):\n");
    yylex();
    return 0;
}

int yywrap() {
    return 1;
}
```

OUTPUT:

```
student@oslab-02:~/220905128/lab5$ ./q6
Enter the C code (Ctrl+D to end input):
#include <stdio.h>

int main() {
    // This is a comment
    int x = 10;
    printf("HELLO WORLD\n");  // ANOTHER COMMENT
    /* This is a multi-line
       comment */
    return 0;
}#include <stdio.h>

int main() {
    // This is a comment
    int x = 10;
    printf("hello world\n");  // ANOTHER COMMENT
    /* This is a multi-line
       comment */
    return 0;
}student@oslab-02:~/220905128/lab5$
```