

NAME: AYANIK PAUL  
Roll No. 22  
D1

## LAB 5

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

CODE:

```
q1.l x q4.l x q5.l x q6.l
1  %{
2  #include <stdio.h>
3  int vowels = 0;
4  int consonants = 0;
5  %{
6
7  %%
8  [aeiouAEIOU] { vowels++; }
9  [b-df-hj-np-tv-zB-DF-HJ-NP-TV-Z] { consonants++; }
10
11  .|\n          {} // Ignore other characters (e.g., spaces, punctuation, etc.)
12  %%
13
14  int main() {
15      printf("Enter a string: ");
16      yylex();
17      printf("Vowels: %d\n", vowels);
18      printf("Consonants: %d\n", consonants);
19      return 0;
20  }
21  int yywrap() {
22      return 1;
23  }
24
```

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 x q2.l x q4.l
1  %{
2  #include <stdio.h>
3  int words = 0;
4  int characters = 0;
5  int blanks = 0;
6  int lines = 0;
7  %}
8
9  %%
10 [ \t]          { blanks++; characters++; }
11 [a-zA-Z0-9]+   { words++; characters += yyleng; }
12 \n             { lines++; characters++; }
13 .              { characters++; }
14 %%
15
16 int main() {
17     printf("Enter text (Ctrl+D to end input):\n");
18     yylex();
19     printf("\nWords: %d\n", words);
20     printf("Characters: %d\n", characters);
21     printf("Blanks (spaces/tabs): %d\n", blanks);
22     printf("Lines: %d\n", lines);
23     return 0;
24 }
25
26 int yywrap() {
27     return 1;
28 }
29
```

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 point number and negative floating point number

CODE:

```
q3.l x q1.l x q2.l x q4.l
1  %{
2  #include <stdio.h>
3  int positive_integers = 0;
4  int negative_integers = 0;
5  int positive_floats = 0;
6  int negative_floats = 0;
7  %}
8
9  %%
10 [0-9]+          { positive_integers++; }
11 -[0-9]+         { negative_integers++; }
12 [0-9]+\.[0-9]+  { positive_floats++; }
13 -[0-9]+\.[0-9]+ { negative_floats++; }
14
15 .|\n           {} // Ignore other characters (spaces, punctuation, etc.)
16 %%
17
18 int main() {
19     printf("Enter the input (Ctrl+D to end):\n");
20     yylex();
21     printf("\nPositive integers: %d\n", positive_integers);
22     printf("Negative integers: %d\n", negative_integers);
23     printf("Positive floating point numbers: %d\n", positive_floats);
24     printf("Negative floating point numbers: %d\n", negative_floats);
25     return 0;
26 }
27
28 int yywrap() {
29     return 1;
30 }
31
```

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 x q3.l x q1.l
1  %{
2  #include <stdio.h>
3  int scanf_count = 0;
4  int printf_count = 0;
5  %}
6
7  %%
8  scanf    { printf("READ"); scanf_count++; }
9  printf   { printf("WRITE"); printf_count++; }
10 .|\n    { ECHO; }
11
12 %%
13
14 int main() {
15     printf("Enter the C code (Ctrl+D to end input):\n");
16     yylex();
17     printf("\nNumber of scanf: %d\n", scanf_count);
18     printf("Number of printf: %d\n", printf_count);
19     return 0;
20 }
21 int yywrap() {
22     return 1;
23 }
24
```

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:

```
q5.l x q4.l x q3.l
1  %{
2  #include <stdio.h>
3  %}
4
5  %%
6  [0-9]+ {
7      int num = atoi(yytext);
8      printf("%d in hexadecimal is: %X\n", num, num);
9  }
10 .|\n { ECHO; }
11 %%
12
13 int main() {
14     printf("Enter the input (Ctrl+D to end input):\n");
15     yylex();
16     return 0;
17 }
18
19 int yywrap() {
20     return 1;
21 }
22
```

OUTPUT:

```
student@oslab-02:~/220905128/lab5$ ./q5
Enter the input (Ctrl+D to end input):
10 11 123 15 0
10 in hexadecimal is: A
11 in hexadecimal is: B
123 in hexadecimal is: 7B
15 in hexadecimal is: F
0 in hexadecimal is: 0
```



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

CODE:

```
q6.l  x  q5.l  x  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$
```