

COMPILER DESIGN LAB 5

📅 Date @February 3, 2026

Name: Devadathan N R
Registration No: 230905010
Class: CSE - A1
Roll No: 04

Question 1

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

CODE:

I5q1.l

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

int vowelTotal = 0;
int consonantTotal = 0;
}%

%%
[aeiouAEIOU]          { vowelTotal++; }
[b-df-hj-np-tv-zB-DF-HJ-NP-TV-Z] { consonantTotal++; }
.|\\n                 { }
%%

int main()
{
    yylex();
    printf("Number of vowels: %d\\n", vowelTotal);
```

```

    printf("Number of consonants: %d\n", consonantTotal);
    return 0;
}

int yywrap()
{
    return 1;
}

```

OUTPUT:

```

CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ echo "Hello World" >
input.txt
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ flex l5q1.l
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ gcc lex.yy.c -o l5q1
-lfl
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ ./l5q1 < input.txt
Number of vowels: 3
Number of consonants: 7

```

Question 2

Count the number of words, characters, blanks and lines in the given input.

CODE:

l5q2.l

```

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

int charCount = 0;
int wordCount = 0;
int lineCount = 0;

```

```

int blankCount = 0;
%}

%%
[a-zA-Z]+    { wordCount++; charCount += strlen(yytext); }
\n          { lineCount++; charCount++; }
[ \t]       { blankCount++; charCount++; }
.           { charCount++; }
%%

int main()
{
    yylex();
    printf("Lines: %d\n", lineCount);
    printf("Words: %d\n", wordCount);
    printf("Characters: %d\n", charCount);
    printf("Blanks: %d\n", blankCount);
    return 0;
}

int yywrap()
{
    return 1;
}

```

OUTPUT:

```

CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ flex l5q2.l
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ gcc lex.yy.c -o l5q2
-lfl
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ ./l5q2
Hello i am Trump
How are u doing
I am the president of USA.
Lines: 3
Words: 14
Characters: 60
Blanks: 11

```

Question 3

Find the number of positive integer, negative integer, positive floating point number and negative floating point number.

CODE:

l5q3.l

```

%{
#include <stdio.h>

int posIntCount = 0;
int negIntCount = 0;
int posFloatCount = 0;
int negFloatCount = 0;
}%

%%
-[0-9]+\.[0-9]+ { negFloatCount++; }
[0-9]+\.[0-9]+ { posFloatCount++; }
-[0-9]+ { negIntCount++; }
[0-9]+ { posIntCount++; }
[ \t\n] { }

```

```

.      {}
%%

int main()
{
    yylex();

    printf("Positive Integers: %d\n", posIntCount);
    printf("Negative Integers: %d\n", negIntCount);
    printf("Positive Floating Numbers: %d\n", posFloatCount);
    printf("Negative Floating Numbers: %d\n", negFloatCount);

    return 0;
}

int yywrap()
{
    return 1;
}

```

OUTPUT:

```

CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ flex 15q3.1
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ gcc lex.yy.c -o 15q3
-lfl
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ ./15q3
10 -5 3.14 -2.7 42 -100 0.5
Positive Integers: 2
Negative Integers: 2
Positive Floating Numbers: 2
Negative Floating Numbers: 1

```

Question 4

Given an 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:

I5q4.I

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

int scanCount = 0;
int printCount = 0;

FILE *yyin, *yyout;
}%

%%
"scanf"    { scanCount++; fprintf(yyout, "READ"); }
"printf"   { printCount++; fprintf(yyout, "WRITE"); }
.|\n      { fprintf(yyout, "%s", yytext); }
%%

int main(int argc, char *argv[])
{
    if (argc != 3)
    {
        printf("Usage: %s <inputfile> <outputfile>\n", argv[0]);
        return 1;
    }

    yyin = fopen(argv[1], "r");
    if (!yyin)
    {
        perror("Input file open failed");
```

```

        return 1;
    }

    yyout = fopen(argv[2], "w");
    if (!yyout)
    {
        perror("Output file open failed");
        return 1;
    }

    yylex();

    fclose(yyin);
    fclose(yyout);

    printf("Number of scanf: %d\n", scanCount);
    printf("Number of printf: %d\n", printCount);

    return 0;
}

int yywrap()
{
    return 1;
}

```

OUTPUT:

```
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ flex l5q4.l
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ gcc lex.yy.c -o l5q4 -lfl
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ cat > input.c
#include <stdio.h>

int main() {
    int x;
    printf("Enter number: ");
    scanf("%d",&x);
    printf("Value = %d", x);
}
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ ./l5q4 input.c output.c
Number of scanf: 1
Number of printf: 2
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ cat output.c
#include <stdio.h>

int main() {
    int x;
    WRITE("Enter number: ");
    READ("%d",&x);
    WRITE("Value = %d", x);
}
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ █
```

Question 5

Write a FLEX program that changes a number from decimal to hexadecimal notation.

CODE:

l5q5.l

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

void printHex(int n)
{
```



```

char hex[20];
int i = 0, r;

if(n == 0)
{
    printf("Hex value: 0\n");
    return;
}

while(n > 0)
{
    r = n % 16;

    if(r < 10)
        hex[i++] = r + '0';
    else
        hex[i++] = r - 10 + 'A';

    n = n / 16;
}

printf("Hex value: ");

for(int j = i-1; j >= 0; j--)
    printf("%c", hex[j]);

printf("\n");
}
}%}

%%
[0-9]+    { printHex(atoi(yytext)); }
.\n      { }
%%

int main()

```

```

{
    yylex();
    return 0;
}

int yywrap()
{
    return 1;
}

```

OUTPUT:

```

CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ flex l5q5.l
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ gcc lex.yy.c -o l5q5 -lf1
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ ./l5q5
10
Hex value: A
25
Hex value: 19
255
Hex value: FF
100
Hex value: 64
512
Hex value: 200
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ █

```

Question 6

Convert uppercase characters to lowercase characters of a C file excluding the characters present in the comments.

CODE:

l5q6.l

```

%{
#include <stdio.h>
#include <ctype.h>
}%

%x CMT

%%
"//".*      { printf("%s", yytext); }

"/*"       { BEGIN(CMT); printf("%s", yytext); }

<CMT>"*/"   { BEGIN(INITIAL); printf("%s", yytext); }

<CMT>.|\\n   { printf("%s", yytext); }

[A-Z]       { putchar(tolower(yytext[0])); }

.|\\n       { printf("%s", yytext); }
%%

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

    yylex();

    return 0;
}

int yywrap()
{

```

```
    return 1;
}
```

OUTPUT:

```
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ flex l5q6.l
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ gcc lex.yy.c -o l5q6 -lf1
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ cat > sample.c
#include <STDIO.H>
```

```
// THIS IS SINGLE LINE COMMENT
```

```
int MAIN() {
    int X = 10;
    printf("HELLO WORLD");
```

```
    /* MULTI LINE
       COMMENT TEST */
```

```
    return 0;
```

```
}
```

```
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ ./l5q6 sample.c
#include <stdio.h>
```

```
// THIS IS SINGLE LINE COMMENT
```

```
int main() {
    int x = 10;
    printf("hello world");
```

```
    /* MULTI LINE
       COMMENT TEST */
```

```
    return 0;
```

```
}
```

```
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/05_LAB$ █
```