

CSE 358 - Assignment 5

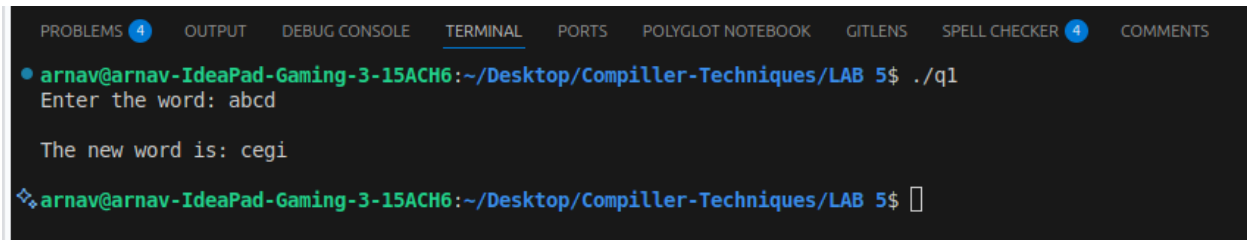
Arnav Jain - 220002018

Question 1

Code:

```
q1.l x
q1.l
1  %{
2      int count = 0;
3      char output[1000];
4      int index_string = 0;
5  %}
6
7  %%
8  [a-zA-Z] {
9      char c = yytext[0];
10     int shift = 2 + count;
11     if (shift > 25) {
12         shift = 1;
13     }
14
15     if (c >= 'a' && c <= 'z') {
16         c = ((c - 'a' + shift) % 26) + 'a';
17     } else if (c >= 'A' && c <= 'Z') {
18         c = ((c - 'A' + shift) % 26) + 'A';
19     }
20
21     output[index_string++] = c;
22     count++;
23 }
24 .|\n { output[index_string++] = yytext[0]; } |
25
26 %%
27
28 int main() {
29     printf("Enter the word: ");
30     fflush(stdout);
31     yylex();
32     output[index_string] = '\0';
33     printf("\nThe new word is: %s\n", output);
34     return 0;
35 }
36
37 // commands to run:
38 // lex -o q1.c q1.l
39 // gcc q1.c -ll -o q1
40 // ./q1
```

Result:



The screenshot shows a terminal window with a dark background. At the top, there is a horizontal menu with several tabs: PROBLEMS (with a blue circle containing the number 4), OUTPUT, DEBUG CONSOLE, TERMINAL (which is currently selected and underlined), PORTS, POLYGLOT NOTEBOOK, GITLENS, SPELL CHECKER (with a blue circle containing the number 4), and COMMENTS. The terminal content shows a prompt 'arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler-Techniques/LAB 5\$' followed by the command './q1'. Below this, the text 'Enter the word: abcd' is displayed. Then, the text 'The new word is: cegi' is shown. Finally, the prompt 'arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler-Techniques/LAB 5\$' is shown again with a cursor, indicating the program has finished execution.

```
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler-Techniques/LAB 5$ ./q1
Enter the word: abcd

The new word is: cegi
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler-Techniques/LAB 5$
```

Question 2

Code:

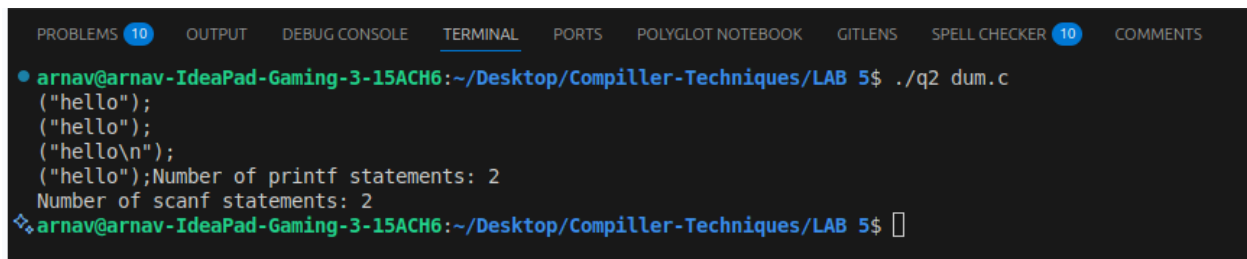
```
q2.l x
q2.l
1  %{
2      int printf_count = 0;
3      int scanf_count = 0;
4      FILE *input_file;
5  %}
6
7  %%
8  "printf"  { printf_count++; }
9  "scanf"   { scanf_count++; }
10
11 %%
12 int main(int argc, char *argv[]) {
13     if (argc < 2) {
14         printf("Usage: %s <input_file>\n", argv[0]);
15         return 1;
16     }
17     input_file = fopen(argv[1], "r");
18     if (input_file == NULL) {
19         perror("Error opening file");
20         return 1;
21     }
22     yyin = input_file;
23     yylex();
24     printf("Number of printf statements: %d\n", printf_count);
25     printf("Number of scanf statements: %d\n", scanf_count);
26     fclose(input_file);
27     return 0;
28 }
29
30 // commands:
31 // lex -o q2.c q2.l
32 // gcc q2.c -ll -o q2
33 // ./q2 dum.c
34
```

Code tested on file dum.c:



```
C dum.c 4 x
C dum.c > [?] scanf
1  scanf("hello");
2  printf("hello");
3  printf("hello\n");
4  scanf("hello");
```

Result:



```
PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS POLYGLOT NOTEBOOK GITLENS SPELL CHECKER 10 COMMENTS
• arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler-Techniques/LAB 5$ ./q2 dum.c
('hello');
('hello');
('hello\n');
('hello');Number of printf statements: 2
Number of scanf statements: 2
❖ arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler-Techniques/LAB 5$
```

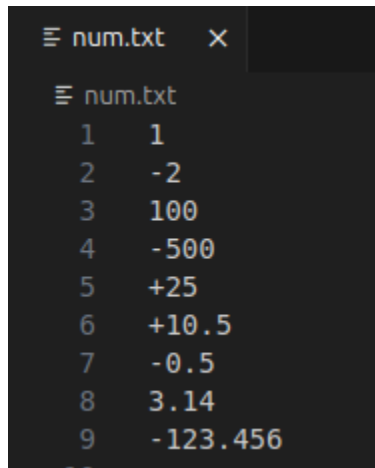
Question 3

Code:

```
≡ q3.l  ×
≡ q3.l
1  %{
2      int pos_int = 0;
3      int neg_int = 0;
4      int pos_frac = 0;
5      int neg_frac = 0;
6  %}
7
8  %%
9  [+ -]?[0-9]+ {
10     if (yytext[0] == '-') {
11         neg_int++;
12     } else {
13         pos_int++;
14     }
15 }
16
17 [+ -]?[0-9]+\.[0-9]+ {
18     if (yytext[0] == '-') {
19         neg_frac++;
20     } else {
21         pos_frac++;
22     }
23 }
24
25 %%
26
```

```
q3.l x
q3.l
25  %%
26
27  int main(int argc, char *argv[]) {
28      if (argc < 2) {
29          printf("Usage: %s <input_file>\n", argv[0]);
30          return 1;
31      }
32      FILE *input_file = fopen(argv[1], "r");
33      if (input_file == NULL) {
34          perror("Error opening file");
35          return 1;
36      }
37      yyin = input_file;
38      yylex();
39      printf("Positive integers: %d\n", pos_int);
40      printf("Negative integers: %d\n", neg_int);
41      printf("Positive fractions: %d\n", pos_frac);
42      printf("Negative fractions: %d\n", neg_frac);
43      fclose(input_file);
44      return 0;
45  }
46
47  // commands:
48  // lex -o q3.c q3.l
49  // gcc q3.c -ll -o q3
50  // ./q3 num.txt
51
```

File on which it was tested:



```
num.txt
1 1
2 -2
3 100
4 -500
5 +25
6 +10.5
7 -0.5
8 3.14
9 -123.456
```

Result:



```
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler-Techniques/LAB 5$ ./q3 num.txt

Positive integers: 3
Negative integers: 2
Positive fractions: 2
Negative fractions: 2
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler-Techniques/LAB 5$
```

For code , refer GitHub

<https://github.com/arnavjain2710/Compiler-Techniques/tree/main/LAB%205>