

# CS 358 LAB 2

Arnav Jain - 220002018

## Question 1

Code:

```
C question1.c X
C question1.c > main()
1  #include <stdio.h>
2  #include <ctype.h>
3
4  int main()
5  {
6      char str[100];
7      int vowels = 0, consonants = 0, i = 0;
8
9      printf("Enter a string: ");
10     fgets(str, sizeof(str), stdin);
11
12     while (str[i] != '\0')
13     {
14         char ch = tolower(str[i]);
15         if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u')
16         {
17             vowels++;
18         }
19         else if ((ch >= 'a' && ch <= 'z'))
20         {
21             consonants++;
22         }
23         i++;
24     }
25
26     printf("Vowels: %d\n", vowels);
27     printf("Consonants: %d\n", consonants);
28
29     return 0;
30 }
31
```

Output:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS POLYGLOT NOTEBOOK GITLENS SPELL CHECKER
• arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler/LAB 2$ ./q1
Enter a string: hello
Vowels: 2
Consonants: 3
• arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler/LAB 2$ ./q1
Enter a string: 12222
Vowels: 0
Consonants: 0
• arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler/LAB 2$ ./q1
Enter a string: arnav
Vowels: 2
Consonants: 3
• arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler/LAB 2$
```

## Question 2

Code:

```
C question2.c x
C question2.c > ...
You, 1 second ago | 1 author (You)
1  #include <stdio.h>
2  #include <string.h>
3
4  int isKeyword(char *str)
5  {
6      char *keywords[] = {
7          "int", "float", "char", "if", "else", "while", "for", "return", "void", "switch",
8          "case", "break", "continue", "default", "struct", "union", "typedef", "long",
9          "short", "unsigned", "signed", "enum", "const", "volatile", "goto", "NULL", NULL};
10
11     for (int i = 0; keywords[i] != NULL; i++)
12     {
13         if (strcmp(str, keywords[i]) == 0)
14         {
15             return 1;
16         }
17     }
18     return 0;
19 }
20
21 int main()
22 {
23     char str[100];
24
25     printf("Enter a string: ");
26     fgets(str, sizeof(str), stdin);
27
28     str[strcspn(str, "\n")] = '\0';
29
30     if (isKeyword(str))
31     {
32         printf("%s is a keyword.\n", str);
33     }
34     else
35     {
36         printf("%s is not a keyword.\n", str);
37     }
38
39     return 0;
40 }
```

Result:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS POLYGLOT NOTEBOOK GITLENS SPELL CHECKER
• arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler/LAB 2$ ./q2
Enter a string: hello
"hello" is not a keyword.
• arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler/LAB 2$ ./q2
Enter a string: int
"int" is a keyword.
• arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler/LAB 2$ ./q2
Enter a string: intx
"intx" is not a keyword.
• arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler/LAB 2$ ./q2
Enter a string: NULL
"NULL" is a keyword.
• arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler/LAB 2$ ./q2
Enter a string: null
>null" is not a keyword.
o arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler/LAB 2$
```

## Question 3

Code:

C question3.c x

```
C question3.c > main(int, char * [])
You, 4 seconds ago | 1 author (You)
1  #include <stdio.h>
2  #include <string.h>
3
4  #define MAX_LINE_LENGTH 1024
5
6  void checkSingleLineComment(char *line) {
7      char *commentStart = strstr(line, "//");
8      if (commentStart != NULL) {
9          printf("Single-line comment: %s\n", commentStart + 2);
10     }
11 }
12
13 void handleMultiLineComment(FILE *file) {
14     char line[MAX_LINE_LENGTH];
15     int isMultiLineComment = 0;
16
17     while (fgets(line, sizeof(line), file)) {
18         char *start = strstr(line, "/*");
19         char *end = strstr(line, "*/");
20
21         if (isMultiLineComment) {
22             printf("Multi-line comment: %s", line);
23             if (end != NULL) {
24                 isMultiLineComment = 0;
25             }
26         } else if (start != NULL) {
27             printf("Multi-line comment: %s", start);
28             isMultiLineComment = 1;
29             if (end != NULL) {
30                 isMultiLineComment = 0;
31             }
32         }
33     }
34 }
35
36
37 int isCommentLine(char *line) {
38
39     if (strstr(line, "//") != NULL) {
40         return 1;
41     }
42     if (strstr(line, "/*") != NULL || strstr(line, "*/") != NULL) {
43         return 1;
44     }
45     return 0;
46 }
```

C question3.c x

C question3.c > main(int, char \*[])

```
37 int isCommentLine(char *line) {
47
48 int main(int argc, char *argv[]) {
49     if (argc != 2) {
50         printf("Usage: %s <file_name>\n", argv[0]);
51         return 1;
52     }
53
54     FILE *file = fopen(argv[1], "r");
55     if (file == NULL) {
56         perror("Error opening file");
57         return 1;
58     }
59
60     char line[MAX_LINE_LENGTH];
61     int lineNumber = 0;
62     int isInsideMultiLineComment = 0;
63
64     while (fgets(line, sizeof(line), file)) {
65         lineNumber++;
66
67         if (isInsideMultiLineComment) {
68             printf("Multi-line comment: %s", line);
69             if (strstr(line, "*/") != NULL) {
70                 isInsideMultiLineComment = 0;
71             }
72             continue;
73         }
74         if (strstr(line, "/*") != NULL) {
75             printf("Multi-line comment: %s", strstr(line, "/*"));
76             isInsideMultiLineComment = 1;
77             if (strstr(line, "*/") != NULL) {
78                 isInsideMultiLineComment = 0;
79             }
80             continue;
81         }
82         if (strstr(line, "//") != NULL) {
83             printf("Single-line comment: %s\n", strstr(line, "//") + 2);
84             continue;
85         }
86         printf("Line %d is not a comment: %s", lineNumber, line);
87     }
88     fclose(file);
89     return 0;
90 }
91
92
```

Examples on which this code was tested on:

```
C example.c 1 x
C example.c
You, 1 second ago | 1 author (You)
1 // hello
2
3 /*
4 hello
5 arnav
6 */
7
8 /arnav
9
10 // arnav
11
12 forrr(int i)
13
14 // arnav
15
```

Result:

```
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS POLYGLOT NOTEBOOK GITLENS SPELL CHECKER 5
• arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler/LAB 2$ ./q3 example.c
Single-line comment: hello

Line 2 is not a comment:
Multi-line comment: /*
Multi-line comment: hello
Multi-line comment: arnav
Multi-line comment: */
Line 7 is not a comment:
Line 8 is not a comment: /arnav
Line 9 is not a comment:
Single-line comment: arnav

Line 11 is not a comment:
Line 12 is not a comment: forrr(int i)
Line 13 is not a comment:
Single-line comment: arnav
○ arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler/LAB 2$
```

To view the code files , refer github

<https://github.com/arnavjain2710/Compiler-Techniques/tree/main/LAB%202>