

Bangladesh Army International University of Science and Technology

Department of Computer Science and Engineering

Lab Report

Lab Report No : 3
Lab Report Name : Write a Program to Scan and Count the Number of Characters, Words and Lines in a File.
Course Title : Compiler Design and Construction Sessional
Course Code : CSE-414
Name : Md Khaled Bin Joha
ID : 0822220105101052
Level : 4 **Term** : 1 **Section** : B **Group** :
Date of Submission : 08/01/26 **Semester** : FALL **Year** : 2026
6 25

Key Learnings:

- Learn file input/output operations in C.
- Implement counters for characters, words (space-separated), and lines (newline-separated).
- Handle edge cases like empty files or trailing spaces.

Code Implementation:

```
Lab Works > C lab3.c > getUserInput(const char *)
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  #include <ctype.h>
5  #include <stdbool.h>
6
7  #define BUFFER_SIZE 4096
8  #define MAX_FILENAME_LENGTH 255
9
10 char buffer[BUFFER_SIZE];
11
12 void clearInputBuffer(){
13     int c;
14     while((c = getchar()) != '\n' && c != EOF);
15 }
16
17 void getUserInput(const char *prompt){
18     printf("%s", prompt);
19     clearInputBuffer();
20
21     if(fgets(buffer, BUFFER_SIZE, stdin) != NULL){
22         size_t len = strlen(buffer);
23         if(len > 0 && buffer[len - 1] == '\n'){
24             buffer[len - 1] = '\0';
25         }
26     }
27 }
28
29 FILE *openFile(const char *filename, const char *mode){
30     FILE *file = fopen(filename, mode);
31     if(file == NULL){
32         perror("Error opening file");
33     }
34     return file;
35 }
36
```

```
Lab Works > C lab3.c > getUserInput(const char *)
29 FILE *openFile(const char *filename, const char *mode){
36
37 void writeToFile(const char *filename){
38     FILE *file = openFile(filename, "w");
39     if(file == NULL){
40         return;
41     }
42
43     getUserInput("Enter text to write into the file:\n");
44
45     fprintf(file, "%s\n", buffer);
46     fclose(file);
47     printf("Data written successfully.\n");
48 }
49
50 void appendToFile(const char *filename){
51     FILE *file = openFile(filename, "a");
52     if(file == NULL){
53         return;
54     }
55
56     getUserInput("Enter text to append to the file:\n");
57
58     fprintf(file, "%s\n", buffer);
59     fclose(file);
60     printf("Data appended successfully.\n");
61 }
62
63 void readAndCount(const char *filename){
64     FILE *file = openFile(filename, "r");
65     if(file == NULL){
66         return;
67     }
68
69     int lineCount = 0, wordCount = 0, charCount = 0;
70     bool inWord = false;
```

Lab Works > C lab3.c > getUserInput(const char *)

```
63 void readAndCount(const char *filename){
64     bool inWord = false;
65
66     printf("\n--- File Content ---\n");
67
68     while(fgets(buffer, BUFFER_SIZE, file) != NULL){
69         printf("%s", buffer);
70         lineCount++;
71
72         size_t len = strlen(buffer);
73         // Not to count new line character
74         if(len > 0 && buffer[len - 1] == '\n'){
75             charCount += len - 1;
76         }
77         else{
78             charCount += len;
79         }
80
81         for(size_t i = 0; i < len; i++){
82             if(isspace((unsigned char)buffer[i])){
83                 inWord = false;
84             }
85             else if(!inWord){
86                 inWord = true;
87                 wordCount++;
88             }
89         }
90     }
91
92     fclose(file);
93
94     printf("\n--- File Statistics ---\n");
95     printf("Lines: %d\n", lineCount);
96     printf("Words: %d\n", wordCount);
97     printf("Characters: %d\n", charCount);
98 }
99
100
101
102
103
104 }
```

```

Lab Works > C lab3.c > getUserInput(const char *)
106 int main() {
107     char filename[MAX_FILENAME_LENGTH + 1];
108     int choice;
109
110     printf("Enter file name: ");
111     if(scanf("%255s", filename) != 1){
112         printf("Error reading filename.\n");
113         return 1;
114     }
115
116     while(1){
117         printf("\n--- File Operations Menu ---\n");
118         printf("1. Write to file\n");
119         printf("2. Append to file\n");
120         printf("3. Read & Count file\n");
121         printf("4. Exit\n");
122         printf("Enter your choice: ");
123
124         if(scanf("%d", &choice) != 1){
125             printf("Invalid input. Please enter a number.\n");
126             clearInputBuffer();
127             continue;
128         }
129
130         switch(choice){
131             case 1:
132                 writeToFile(filename);
133                 break;
134             case 2:
135                 appendToFile(filename);
136                 break;
137             case 3:
138                 readAndCount(filename);
139                 break;
140             case 4:
141                 printf("Exiting program...\n");

```

```

139                 break;
140             case 4:
141                 printf("Exiting program...\n");
142                 return 0;
143             default:
144                 printf("Invalid choice. Please try again.\n");
145         }
146     }
147 }

```

Input Sample:

./lab3

Enter file name: data.txt

--- File Operations Menu ---

1. Write to file

2. Append to file

3. Read & Count file

4. Exit

Enter your choice: 1

Enter text to write into the file:

Hello, I'm calling first function (Write to file)

Output Sample:

```
joha546@joha546:~/Projects/Compiler-Design-and-Construction/Lab Works$ ./lab3
Enter file name: data.txt

--- File Operations Menu ---
1. Write to file
2. Append to file
3. Read & Count file
4. Exit
Enter your choice: 1
Enter text to write into the file:
Hello, I'm calling first function (Write to file)
Data written successfully.

--- File Operations Menu ---
1. Write to file
2. Append to file
3. Read & Count file
4. Exit
Enter your choice: 4
Exiting program...
```