

Assignment 1 – Revisiting File Handling

This is the **introductory level** assignment. This means it has low impact on your grade. You are however required to submit it.

Assignment Statement:

This assignment will be a revision to the concepts and implementations of file handling and that in C programming language. We shall see some *must know* and basic functions.

The basic functions are:

- 1) Opening/Creating a file
- 2) Closing a file
- 3) Reading a file
- 4) Writing in a file

In C language, we use a structure **pointer of file type** to declare a file.

FILE *fp;

The table below provides a list of functions you should know in order to perform the basic tasks listed above:

FUNCTION	USAGE
fopen()	create a new file or open an existing file
fclose()	closes a file
fgetc()	reads a character from a file
fputc()	writes a character to a file
fputs()	writes a line of characters to a file
fgets()	Reads a line of characters from a file
fscanf()	reads a set of data from a file
fprintf()	writes a set of data to a file
getw()	reads a integer from a file

putw()	writes a integer to a file
fseek()	set the position to desire point
ftell()	returns current position of file

1) ***fp = FILE *fopen(const char *filename, const char *mode);**

Following are the modes for fopen:

- i. **r or rb**
Open file for reading.
- ii. **w or wb**
Truncate to zero length or create file for writing.
- iii. **a or ab**
Append; open or create file for writing at end-of-file.
- iv. **r+ or rb+ or r+b**
Open file for update (reading and writing).
- v. **w+ or wb+ or w+b**
Truncate to zero length or create file for update.
- vi. **a+ or ab+ or a+b**
Append; open or create file for update, writing at end-of-file.

****Note: “b” is for binary files**

- 2) **int fclose(FILE *fp);**
- 3) **int fgetc(FILE *stream)**
- 4) **int fputc(int c, FILE *stream)**
- 5) **char* fgets(char *s, int n, FILE *stream)**
- 6) **int fputs(const char *s, FILE *stream)**
- 7) **int fscanf(FILE *stream, const char *format [, argument, ...])**
- 8) **int fprintf(FILE *stream, const char *format [, argument, ...])**
- 9) **int getw(FILE *fp);**
- 10) **int putw(int number, FILE *fp);**
- 11) **int fseek(FILE *stream, long int offset, int whence)**
- 12) **long ftell(FILE *stream)**

Your Tasks:

Following tasks are to be done:

- 1) Write a C program that accepts a file name and a substring as command line argument and prints the number of occurrences of substring in the given file on the console. If the file does not exist, print some error on the screen.
- 2) Write a program to manage record of student registration on file. Enter your details and the details of two of your classmates. Roll no, Name and Email should be saved as student details. Example Record:

Name: Waleed Akram Khan

Roll No. L17-4268

Email: 1174268@lhr.nu.edu.pk

Take input via command line argument or user input.

Your program should do the following:

- a. Allow Adding students
 - b. Read student record from file. (Program must take input from user to read a record. E.g. "0" to read your record, "1" and "2" to read record of fellow student.)
 - c. Delete student record in file. (Program must take input from user to delete a record. E.g. "0" to delete your record, "1" and "2" to delete record of fellow student.)
- 3) Write a program that reverses all the characters of words that are not vowels in a given file.

NOTE: Task 1 and 3 should accept a command line argument which will indicate which file to use for answer validation. For example, the input to task 1 can be something like this:

```
> ./program.out a
```

OR

```
> ./program.out b
```

Where "a" and "b" indicate files *q1_a.txt* and *q1_b.txt* respectively as input that are provided in the zip file.

GitHub Submissions:

This assignment is to be submitted on Github only.

Make sure you add the Teaching Assistant as collaborator on each assignment.

Username: waleedakramkhan

For any queries, consult the TA: Mr. Waleed Akram Khan via email: 1174268@lhr.nu.edu.pk