

File I/O

Goals

- Gain experience with file I/O in C++

In this lab, we will implement a program that reads a file filled with letters, count the letter frequencies in the file, and write the results to a new file.

Requirements

Input File

For simplicity, you can download this example file as the input file: [file_fun.txt](#) 

[Download file_fun.txt](#)

, or, you can also create your own input files.

File should consist of only 'a' - 'z', 'A' - 'Z', **space character, newline characters.**

Program

Your program should first ask for the file the user would like to read, then read the file chosen by the user. The program will continue counting letters until the EOF (end of file) character. **For each paragraph** in the input file, the program should count the frequency of each letters, **ask user what filename** to output to, then proceed to output the frequencies to the file. Each paragraph should have its own output file.

Note: The letter counting **disregards the upper/lower case differences**. This means, capitalized letter 'F' is treated like 'f', and counted toward frequency of 'f'.

Note: the paragraphs are **separated by newline characters**.

So, the following input file means 3 paragraphs: (not counting the quotation mark)

"Hello I am a CS major

I am having fun

This is a placeholder"

The **format** of the output file is as follows:

a: 3

b: 0

c: 1

d: 0

.....

z: 0

Functions

The following is the functions of the program. You can make more function if you would like.

- void count_letters(ifstream &, int*);
- void output_letters(ofstream&, int*);

The requirements of each functions are specified below.

void count_letters(ifstream &, int*)

count_letter function takes an input file stream variable, and a pointer to an array of integers.

The function first reads the paragraph string from the input file stream, then counts the letter frequencies of that paragraph, and store the frequencies in the array. Disregards upper/lower case letters.

void output_letters(ofstream &, int*)

output_letters function takes an output file stream variable, and a pointer to an array of integers that contains frequencies of letters.

The function first asks user for the filename the user would like to output to, then outputs the frequencies of letters to the output file in the format specified above.

File I/O

If the program fails to open input file, then the program can either terminate, or recovers from error and re-prompt user for a new file name.

What to Submit:

- All the program files including header and source files (.cpp/.hpp)
- Input text file
- Makefile

Important: Put all the files in a single .zip file and submit it on Canvas.

Grading:

- Programming style – 10%
- Read in input file correctly – 20%
- Implement count_letters() function correctly – 20%
- Implement output_letters() function correctly – 20%
- Create and write to the output file correctly – 30%