

CSE 5462: Lab 1 (Total: 100 points)

Demo in CL 112: Sep 1 (Thursday)

Electronic Code Submit Deadline: 9pm, Sep 1 (Thursday)

Important: The demonstration schedule will be posted in the discussion group closer to the deadline. The submitted code will be used only to verify that you did not copy from others, to compile and re-run your program, to make sure you were indeed demonstrating your own code, and to grade for documentation of your code.

Write a program in C called “count” to read a binary file and print the following statistics on the screen as well to an output file:

- the size of the file in bytes
- number of times the search-string specified in the second argument appeared in the file

You will run the program using the following command:

`count <input-filename> <search-string> <output-filename>`

The string matching algorithm does not have to be the most efficient in time.

If the input-filename is incorrect, or the number of arguments are incorrect, or the output-file cannot be created, the program must print appropriate messages and show how to correctly invoke it.

The search string will be 20 bytes or less. The search string and the file both may contain binary characters.

The input file can be of any arbitrary length. The implication is that you cannot have the entire file in the memory at any point of time. You will read the file in chunks of 100 bytes or smaller. Note that you also have to handle the case when your search string starts in one chunk and ends in the next chunk.

If the output file already exists, it will overwrite that file.

During execution, any array in the program or malloc-ed space must not exceed 100 bytes.

Submit **well-documented and well indented** code along with a README file explaining how to run the program, and a Makefile, using the following command:

`submit c5462aa lab1 <code-directory-name>`

The grading rubric is as follows:

- Program correctness and robustness: 80
- Coding style (comments, indentations, README, Makefile): 20

FAQ

Q: Which function calls can I use for finding the size of a file?

A: You can use `fstat`, `ftell`, etc.

Q: How do I change my default shell?

A: Use **chsh**.

Q: How do I provide binary characters in the argument?

A: For bash shell:

`count inputfilename $'\x02\x05\xd8' outputfilename`

Q: Will the input file contain only one line of characters or it may contain multiple lines of it?

A: Your program should be able to handle binary files. So multiple lines are fine in the input file. In fact, end

of line character may be present in the search string itself.

Q: Can a character in a stream be matched twice?

A: Yes. If the input file contains 4 characters – “aaaa” and we want to search “aa”, it should return 3.

Q: How much commenting is expected?

A: Like in any professional code, each function must be explained in plain words. Each major step in the code must be explained in words. In your README file write a short description of what the program does, how to compile it and how to run it.

Q: How do I know that the file was transferred correctly.

A: Use “diff” or “md5sum”.

Q: Can I submit multiple times?

A: When you execute submit for lab1, the submissions from all the prior executions of submit (for lab1) are erased. So you should submit all your files with your submit command. You can either list the names of the files or provide the name of the directory where your code (with README and Makefile) resides. You can submit any number of times till the deadline.

Q: Are there any specific instructions on what is expected when compiling the program?

A: Make sure your code does not produce any warning messages when compiling. If you use “-Wall”, make sure you address all the warning messages. In your Makefile have provision for “make clean”.