

Project 4: Working with a file

40 Possible Points

4/4/2022

Attempt 1



IN PROGRESS

Next Up: Submit Assignment



Add Comment

0

Unlimited Attempts Allowed

▼ Details

In this project you will be writing a C program to take in some command line options and do work on input from a file. This will require using command line options from **getopt** and using file library calls in C.

Keep in mind this is a project in C, not in Bash script! If you submit a bash script doing the work, you will receive a 0.

In particular, your program should consist of a file **findc.c** and a header file for it called **findc.h**, as well as a **Makefile** that compiles them into an executable called **findC**.

This executable findC takes the following **optional** command line options:

-h : This should output a help message indicating what types of inputs it expects and what it does. Your program should terminate after receiving a -h

-f filename : When given -f followed by a string, your program should take that filename as input.

-c char : Specifies a different character to look for in the target file. By default this is the character 'c'.

Our program can be run in two ways:

1) Given a file as input by running it with the optional command line argument -f and a filename as input. For example, suppose we had a file with some strings called **inputfile**

```
./findC -f inputfile
```

2) Redirecting input to it as follows:

```
./findC < inputfile
```

So what task is our program doing? Our program will check each line of its input to find out how many 'c' characters the file input or stdin has (or a different character, if the -c command line argument is



(<https://umsystem.instructure.com/courses/97754/modules/items/4386153>)

Assignment

(<https://umsystem.instructure.com/courses/97754/modules/items/4386153>)

where X is the number of c's found in the file.

Note that the project MUST include a Makefile in order to compile the project. If it does not, it will not be graded until you submit one.

As with previous projects, I suggest doing the project incrementally.

- 1) Implement your source files findc.c and findc.h and write a Makefile to make an executable (at this stage, these files can be almost empty shells that just compile correctly).
- 2) Implement the command line arguments using **getopt**, but just have the arguments printed out to the screen. Make sure these options work correctly.
- 3) Now use the command line arguments, if given, to open a file. Output that file to the screen to make sure it is reading the file correctly. Ensure at this stage that if the file doesn't exist or is not readable, that you output error messages.
- 4) Now do your character processing to count how many characters of the particular type are in the file.

Please keep in mind that your program should give error messages (and not just terminate or segfault) if something goes wrong. For example, if given a -f without being followed by a filename, or if given a file that is not readable or does not exist.

Choose a submission type

