

This C bootcamp project is due at 11:59:59pm on September 15, 2014 and is worth 1.25% of your grade. You must complete it by yourself, and no slip days may be used.

1 Description

The goal of this assignment is to give you experience with debugging C code, topics that you will be using later in the semester in your course projects. Your assignment is to complete three C problems. For each of the problems, we give you completed C code that has bugs; your job is to use the debugging tools we learn about in class to locate and fix the bugs. You should not need to add any helper functions (all of the changes should be one-liners), and do not change any of the main functions.

1.1 Starter code

The starter code for the assignment is available in `/course/cs3600f14/code/c-project4`. You must use this code as a basis for your project. Provided is a `cp4-problem[1,2,3].c` code and a `Makefile`. The `Makefile` is configured to correctly compile your code and to test your code on sample input.

To get started, you should copy down this directory into your own local directory. You should first `ssh` over to any CCIS Linux machine (e.g., `bubbles.ccs.neu.edu`). We recommend doing the following, making a directory in your CCIS home directory for all of your CS3600 work, and then copying down the C Bootcamp Project 1 code:

```
bash$ cp -r /course/cs3600f14/code/c-project4 ~/cs3600
bash$ cd ~/cs3600/c-project4
```

2 Completing your project

This project consists of three problems, each located in `cp4-problem[1,2,3].c`. Each file contains documentation for what is expected, and you should only add code where a `TODO` line exists. If you are unsure of any specifications (e.g., what to do in corner cases), first consult the documentation before asking on Piazza. Instructions for compiling, running, and testing your code is the same as C bootcamp project 1.

3 Implementation hints

You should develop your client program on the CCIS Linux machines, as these have the necessary compiler and library support. You are welcome to use your own Linux/OS X machines, but you are responsible for getting your code working, and your code *must* work when graded on the CCIS Linux machines. If you do not have a CCIS account, you should get one ASAP in order to complete the project. Your code must be `-Wall` clean on `gcc`. Do not ask the TA for help on (or post to the

forum) code that is not `-Wall` clean unless getting rid of the warning is what the problem is in the first place.

You will want to make sure that you break your program up into modules, such that each module represents a sensible type abstraction. If you have questions on how to do this, please come see me or the TA.

4 Submitting your project

Once you are done, you should submit your (thoroughly documented) code. You can do so by running the `/course/cs3600f14/bin/turnin` script. Assuming you created a `c-project4` directory, you can simply run

```
/course/cs3600f14/bin/turnin c-project4 ~/cs3600/c-project4
```

If you placed your code in a different location, you can turn it in by running

```
/course/cs3600f14/bin/turnin c-project4 <dir>
```

Where `<dir>` is the name of the directory with your submission. Regardless, when run, you should see output that looks like

```
bash$ /course/cs3600f14/bin/turnin c-project4 ~/cs3600/c-project4
  Added file Makefile (157 bytes)
  Added file cp4-problem1.c (1392 bytes)
  Added file cp4-problem2.c (2163 bytes)
  Added file cp4-problem2.c (3394 bytes)
Successfully submitted c-project4 for user amislove (confirmation ZiWKE5).
Submitted a total of 4 files (8469 bytes) in 1 directories.
```

The script will print out every file that you are submitting, make sure that it prints out all of the files you wish to submit. Finally, make sure you see the “Successfully submitted” link at the end. If you receive any other error messages, email the instructor with the error message if you are not able to diagnose the problem.

5 Advice

A few pointers that you may find useful while working on this project:

- Check the Piazza forum for question and clarifications. You should post project-specific questions there first, before emailing the course staff.
- Finally, get started early and come to the TA lab hours - these are held in the lab at 102 West Village H. You are welcome to come to the lab and work, and ask the TA any questions you may have.