

Assignment 2: C Basics

Assigned: Friday, 3 September 2010

Due: 11:00 p.m., Thursday, 9 September 2010

Summary: In this assignment, you will begin exploring the C programming language through a few simple examples.

Purposes: To get you started thinking about C.

Expected Time: Two to three hours.

Collaboration: Each student should turn in his or her own responses to this assignment. However, you may certainly work with other students as you review the course materials and answer the questions. You may discuss the assignment with anyone you wish. You may obtain help from anyone you wish, but you should clearly document that help.

Note: Because K&R is an important book, answers to all of these exercises are readily available online. My academic honesty guidelines do indicate that you can copy any answers you find online. However, you will find that you learn much more if you do the work on your own.

Submitting: Email me your answers. See further details below.

Warning: So that this assignment is a learning experience for everyone, I may spend class time publicly critiquing your work.

Assignment

Do exercises 1-3, 1-4, 1-7, 1-8, 1-13, and 1-15. In your answer to 1-13, you may group together words of ten or more characters.

Test each of your answers to show that the code works to your satisfaction.

Make sure that you follow the GNU Coding Standards (which are not the standards that K&R follow) in formatting your code.

Important Evaluation Criteria

I will primarily evaluate your programs on their correctness. I will also look at the quality of the tests that you write.

Submitting Your Homework

Please submit this work as an attachment to an email message entitled should be titled CSC161 Assignment 2. Your attachment should be a gzipped tarball that will expand to the directory `hw2.username`.

For each problem on the assignment, you are to write and test a C program. You should submit both your source code and a test script that shows your test results. Please do not put executables in the tarball.

To produce a test script, use the Linux command `script` as follows:

In the terminal window, type:

```
script filename
```

This will begin a script session, the output of which will be written to *filename*. Be careful to use different filenames for each script you want to submit. I would suggest that you use `foo.script` as the script for `foo.c`.

Now continue to test your program (ie, run your program and enter test data for each of your test cases) in the same terminal window. Note that it will make your script easier to read if you enter a few Enter/Return characters between each test.

When you have completed your test runs, type `exit` to finish the script session. At this point all of your interaction during the session, including both input and output, will be saved into the script file.

Unfortunately, a few messy characters will also be entered into your script file for any special function keys, such as arrows, delete, backspace, and home, that you use. It would be nice if you could avoid these keys while producing your script, but I understand that it is hard to avoid them completely.

Please produce one test script for each problem on the assignment. You can look at the test script using either `cat` or `less`, but you must not edit your test script in any way.

Citations

The instructions for creating a typescript are taken from similar instructions by Marge Coahran.
