

CSCI 3215 (Advanced UNIX Command-Line Tools), Fall 2020

Homework 6

Credit: 20 points.

1 Reading

Be sure you have read, or at least skimmed, the assigned readings for 10/14.

2 Programming Problems

Do the following programming problems. You will end up with at least one code file per problem. Submit your program source (and any other needed files) by sending mail to my TMail address with each file as an attachment. Please use a subject line that mentions the course and the assignment (e.g., “csci 3215 hw 6” or “UNIX hw 6”). You can develop your programs on any system that provides the needed functionality, but I will test them on one of the department’s Linux machines, so you should probably make sure they work in that environment before turning them in.

`make`

1. (20 points) Do one of the following:

- Suppose you have a C program consisting of the following files:
 - `defs.h` containing macro and constant definitions.
 - `foo.h` containing constants and prototype definitions for function `foo`.
 - `foo.c` containing code for function `foo`.
 - `bar.h` containing constants and prototype definitions for function `bar`.
 - `bar.c` containing code for function `bar`.
 - `main.c` containing code for the main program, which calls functions `foo` and `bar`.

([Files.zip](#) contains all files.)

Write a makefile that:

- When you type `make main`, creates executable `main` from the source files, compiling just the parts of this program that need to be recompiled (because either source code has changed or an `#include`’d file has changed). Compile using any C compiler (`cc` or `gcc`) but using the flags `-Wall`, `-pedantic`, `-std=c99`, and `-O`.
- When you type `make clean`, deletes all the object files for the program.
- When you type `make xclean`, deletes all the object files for the program and the executable.

You can make use of the implicit rules defined by `make` if they will help. (They probably will.) Try to make good use of makefile variables to reduce duplication.

Be advised that the command `touch` can be used to change a file’s timestamp without opening it in an editor, etc. (This may be helpful in testing.)

- If you have worked with `make` before and would have no trouble with the preceding problem, write your own problem for this assignment: Think of something interesting or useful you could do with `make` and write a makefile that will do it. (You should plan to come up with something you can finish in an hour or two.) Include comments at the top of the makefile describing what is meant to do.

3 Pledge

Include the Honor Code pledge or just the word “pledged”, plus *at least one of the following* about collaboration and help (as many as apply).¹ Text *in italics* is explanatory or something for you to fill in. For programming assignments, this should go in the body of the e-mail or in a plain-text file `pledge.txt` (no word-processor files please).

- This assignment is entirely my own work. (*Here, “entirely my own work” means that it’s your own work except for anything you got from the assignment itself — some programming assignments include “starter code”, for example — or from the course Web site. In particular, for programming assignments you can copy freely from anything on the “sample programs page”.*)
- I worked with *names of other students* on this assignment.
- I got help with this assignment from *source of help — ACM tutoring, another student in the course, the instructor, etc.* (*Here, “help” means significant help, beyond a little assistance with tools or compiler errors.*)
- I got help from *outside source — a book other than the textbook (give title and author), a Web site (give its URL), etc..* (*Here too, you only need to mention significant help — you don’t need to tell me that you looked up an error message on the Web, but if you found an algorithm or a code sketch, tell me about that.*)
- I provided help to *names of students* on this assignment. (*And here too, you only need to tell me about significant help.*)

4 Essay

Include a brief essay (a sentence or two is fine, though you can write as much as you like) telling me what if anything you think you learned from the assignment, and what if anything you found interesting, difficult, or otherwise noteworthy. For programming assignments, it should go in the body of the e-mail or in a plain-text file `essay.txt` (no word-processor files please).

¹ Credit where credit is due: I based the wording of this list on a posting to a SIGCSE mailing list. SIGCSE is the ACM’s Special Interest Group on CS Education.