CSCI 3215 (Advanced UNIX Command-Line Tools), Fall 2022
Reading Quiz 4

Credit: 20 points.

1   Reading

Be sure you have read, or at least skimmed, the reading for 10/03, and also the lecture notes for 10/03 and 10/10.

2   Instructions

For most courses I ask that you answer questions on reading quizzes using *only* the assigned readings for the course. For this course, however, while I mean for you to be able to answer the questions based on the assigned readings, plus sometimes `man` and `info` pages, you can consult any source you like. (So, Web searches are AOK.) It's okay to talk to classmates about this assignment as you usually do, *but* I want each person to at least skim all the reading. Include the Honor Code pledge in what you turn in, either the full pledge or just the word "pledged". For these quizzes by doing this you are also saying you have at least attempted or skimmed all the reading it covers.

Please put the pledge in the same document as your answers, so I don't overlook it, and please be sure to include your name somewhere in the file, so when I print it for grading I know whose work it is.

You may write out your answers by hand and scan them, or you may use a word processor or other program, but please submit PDF or plain text in the "turn-in" folder I have set up for you on Google Drive. (So, no word-processor files, including Google Docs, and no links to other Google Docs.)

3   Questions

1. (2.5 points) Shell scripts can take arguments. Functionality is much like the arguments to `main()` in C or C++ (`int argc` and `char *argv[]`). In C or C++, you check that the user supplied the right number of arguments by testing `argc`. How do you do that in a `bash` script?

2. (2.5 points) What do the following commands do in `bash`?

   ```
   foobar=$(echo $USER | tr [a-z] [A-Z])
   echo $foobar
   ```

   (Here I'm looking not only for what prints, but why — e.g., what does that `$(...)` do? Not sure your answer is what I had in mind? Ask. E-mail works.)

3. (2.5 points) How in `bash` do you check to see whether a file exists?

4. (2.5 points) How in `bash` do you print a list of the first n squares, (i.e., 1, 4, 9, ...n^2)? Assume n is a shell variable previously set (e.g., with `n=10`).

5. (2.5 points) If you want to loop over all files that match a wildcard, you might write

   ```
   for f in $(*.c)
   ```

   Will this always do something sensible?

6. (2.5 points) What's the difference between

   ```
   ls a.txt ; ls b.txt
   ```

   and

   ```
   ls a.txt && ls b.txt
   ```

   or is there one?

7. (2.5 points) What does `shift` do in a shell script?

8. (2.5 points) How does `bash` know/decide what type a variable is?