

# CS 306 – Linux/UNIX Programming – Spring 2012

## Homework #1

(Due: Wednesday 2/1/12)

(Questions that refer to the characteristics of a particular user and/or system are to be answered based on what is found using *your* account on the CS Dept. Linux workstations. Some questions will not be covered completely in class/text, requiring that you read man pages or other documentation—just as you would have to do using Linux outside of class.)

1. What is your (1) *username*, (2) default *group*, (3) *UID*, and (4) *GID*?
2. What is the (absolute) *path* of your *home* directory?
3. What are *four commands* that will reset the *current working directory* to your home directory, regardless of what directory you are currently in? (Hint: all involve `cd`.)
4. What is the best command to determine the *path* of the `ls` command/program, and what is `ls`'s path?
5. Suppose that your *current working directory* is `~/tmp/code/306`, and you want to make `~/courses/cs306` the current working directory. Give two different commands to do this, one using an *absolute* path and the other a *relative* path.
6. A key (local) configuration file for the bash shell is `~/.bashrc`. Why isn't this file listed when you run the `ls` command in your home directory, and how could you get it listed using `ls`?
7. By default, when you give a directory as an argument to `ls`, the *contents* of the directory are listed rather than the directory itself. What *option(s)* would you give to `ls` so that you get a "long listing" of the argument directory *itself*?
8. Give a command to *move* all of the *C source files* (with standard format names for GCC) from the directory `~/cs306` to the directory `~/cs306/labs` (that you have already created). Do not make any assumptions about the CWD.
9. Suppose that you want to *copy* a file named `test` in the *current working directory* to your *home directory*, but you want to be *prompted* if this is going to cause a file to be *overwritten*. What command (including *option(s)*) would you run to do this?
10. How could you make certain that `cp` is *always* going to be run with the option(s) you identified in the previous question without you having to always type it? Explain exactly what you would have to do to enable this.
11. How would you check the default *search path* for your account, and what is it? (List no more than the first *five* entries.)
12. How would you check if the *current working directory* is *always* in the default search *path*, and is it?
13. Give a (single) *command* that will *print* out the *number* of files (files in a generic sense) that are in the current working directory. (Hint: requires *piping*.)
14. Give a (single) *command* that will *print* out all the *lines* in the file `~/cs03/lab1/lab1.c` that contain calls of the `printf` *function*. Do not make any assumptions about the CWD.

[OVER]

15. Give a (single) *command* that places a list of all of the files in the current directory that were last *modified in July* into the file `~/july-files`. (Hint: use piping with one or more filters, plus redirection.)

(This is actually a surprisingly difficult problem due to: (1) variable spacing in `ls`' output and (2) the possibility of whitespace in filenames. Because of this, you are allowed to end up with more than just the filenames in the output file (as long as the relevant filenames are clear). If you want to try to figure out how to end up with just the filenames, that will be worth *extra credit*—but be sure your method works for filenames containing whitespace! I would probably use AWK to do this, but former CS 306 student Tim Hansom found a quite elegant solution using only basic Linux commands in a pipeline.)

16. List the files in the following listing that would be *deleted by the command*:
- ```
rm ?{ab,cd,ef}*-[2-4].dat?
```
- (note that the ?'s are part of the command filename pattern)

```
abcdeh-3.data  ibcde-2.dat1      nbefg-3.dat2
dab-2.data     icd2-5-3.data     pabcdf3-4.data
eabc-2.data    icd2ef1.data      pef3.data1
eefg-3.data    jcd2ef-3.dat      pefcd-34.data
iab7-4.data    jcd2ef-3.data     pefcd-3.data
iab32.dat      jcef21-23.dat     pefcd-3.save
```

17. Suppose you want to *move* a large number of files from the directory `~/tmp` to the directory `~/save`, and that the relevant filenames have this *pattern*: they begin with “306”, contain “info”, and have a single extension of “txt” or “text”. What is a *single command* with appropriate filename *pattern* that will do this move for you?
18. Suppose that the command `ls -l test` produces the line:
- ```
-rw-r-x--x    2 carver  faculty      1732 Jan 23 20:39 test:
```
- (a) Who is the file's owner and what are his/her access permissions?  
(b) What is the group associated with the file and what are its access permissions?  
(c) What users have permission to *read test*?  
(d) What size is the file?  
(e) What does the Jan 23 20:39 mean?
19. Show *two commands* that one could use to change the access permissions on (the above mentioned file) `test`, so that `carver` has read and write access only, (other) members of `faculty` have read access only, and all other users have no access.
20. What is the *simplest command* for adding *write* permissions to `test` for *all users* (without changing any other permissions)?
21. Show *two different versions* (i.e., different options) of the `ps` command, that will show *all processes* on a system, and show (at least) the following information about each process: (1) PID, (2) owner/UID, (3) the command being run, (4) start time/date, and (5) total CPU time.
22. Give a *command* that would *definitely terminate a process* whose *PID* is 4513 (assuming the user has permission to terminate the process).