CSC 180 lab 1: Familiarization

Mon Sept 10 or Thurs Sept 13, 2001

This first lab session is merely to familiarize you with the lab; it does not involve producing anything and it is not going to be graded. Experienced computer users can probably get through it very quickly, but you probably will still learn a thing or two and should still complete the lab.

The lab rooms for CSC 180 are SF1012, SF1106, and EA109. You will see on your timetable that you have some lab time scheduled (the ‘‘practical’’ section). This is not the only time you can work in the labs; it’s just when they’re booked for our course and when a TA will be there (ideally—at a minimum I should manage to arrange to have one TA going between SF1012 and SF1106).

There is a total of twelve lab sessions. The first lab session is Monday September 10th or Thursday September 13th (depending on section registration); there is no lab session on Monday October 8th, which is a holiday; the last lab session is Monday December 3th or Thursday November 29th.

Since the ‘‘labs’’ (as opposed to ‘‘assignments’’) are not going to be graded, you may discuss them freely with other students, and you are encouraged to do so to some extent, although you should be sure that you do end up doing the lab. However, in general in computer programming (and in many other things), it is far easier to recognize a valid solution than to create that solution yourself; reading someone else’s lab solution is *not* a replacement for doing the lab yourself.

1. Find a working, available computer and ‘‘log in’’.

Your username (or account name, user id, login name, or logname—all of these terms mean the same thing) is your surname, not capitalized, up to 7 letters long; if that’s less than 7 letters long then you might have to append your initials. There is a program which finds your username; to run it, instead of your username type ‘‘getname’’, and just press return if it asks for a password.

Your initial password is an eight-digit string comprised of the last four digits of your student number, then your birth month as two digits (e.g. March is 03), then your birth day number as two digits (e.g. March 5th would give you 05).

If you are in a ‘‘PC lab’’ (in MC or WB), you need to run a program called TeraTerm SSH to connect to the unix/linux computers. Normally you should use one of the unix labs (SF1012/SF1106/EA109).

(I use the term ‘‘unix’’ inclusively below, to include linux—there are many varieties of UNIX™, and the primary reason linux isn’t one of them is the ™ part, not the UNIX part.)

2. If you haven’t used a mouse before, start getting used to it now. (It might take a while until you are really comfortable with it; some people find it very intuitive and some don’t.) Click with the left mouse button to select things. Experiment with moving windows around, changing their size, raising them (so that they seem to be on top of the other windows, in the places where they overlap), and directing the ‘‘focus’’ (which controls into which window your typing goes).

3. Change your password by typing ‘‘passwd’’ at a shell prompt. Why does it ask for your new password twice? Why does it prompt for your old password first?

4. To create and modify your computer programs you will have to use an application program called an ‘‘editor’’. You can use any available editor you like, but a simple one to start with is called ‘‘pico’’. Type ‘‘pico hello.c’’ to begin editing a new file called ‘‘hello.c’’.

5. Here is a little C program. Type this in as the contents of hello.c, and save it, and exit pico. Pay attention to all of the punctuation symbols; they are crucial. For one, note that there is a difference between the ‘‘curly braces’’ (set brackets) and the parentheses.

```
int main()
{
    printf("Hello, world\n");
    return 0;
}
```

(over)

6. The ‘‘cat’’ command is the easiest way to display a short file on the screen in unix. Type ‘‘cat hello.c’’ to see your new file. To see a list of your files so far, type ‘‘ls’’.

7. To translate this C program into a machine language program so that you can run it, you need to run the C compiler, called ‘‘cc’’. Type ‘‘cc hello.c’’. If you do not get any error messages, you will now have a compiled version of this program called ‘‘a.out’’. Type ‘‘ls’’ to check.

8. Run it by typing ‘‘`./a.out`’’ (n.b. this is a ‘‘forward slash’’—MS-DOS’s use of the backslash here is idiosyncratic). (Simply ‘‘`a.out`’’ may or may not work; more about this next week.)

9. You can also save the output into a file with a greater-than symbol; think of it as an arrow. Type ‘‘`./a.out >x`’’ to ‘‘redirect’’ the output to a file named ‘‘x’’. Type ‘‘ls’’ to see a list of all the files you have by now. Type ‘‘cat x’’ to see the contents of the file ‘‘x’’.

10. Type ‘‘mv x y’’ (for ‘‘move’’) to change the name of the file ‘‘x’’ to ‘‘y’’. Then try typing ‘‘ls’’, ‘‘cat y’’, and ‘‘cat x’’. Type ‘‘rm y’’ (for ‘‘remove’’) to delete the file. Type ‘‘ls’’. Also try the ‘‘cp’’ command (‘‘copy’’), which is like the ‘‘mv’’ command except it creates a new file, so that then there are two.

11. As with editors, there are several available e-mail reading programs. The mail reader called ‘‘pine’’ is a good but simple one, and it has the same pico editor built in. Type ‘‘pine’’ to start it. Beware its habit of displaying important messages at the bottom.

Your own e-mail address consists of your user name, an ‘at’ sign, and then ‘‘ecf.utoronto.ca’’. For example, if your user name is ‘‘betty’’, then your e-mail address is ‘‘betty@ecf.utoronto.ca’’. Send an e-mail message to a friend. Read an e-mail message sent by a friend.

12. You will have heard of (and likely have used) the ‘‘world-wide web’’. Further labs will not be distributed on paper, only on web pages. There are two versions of each lab handout: an HTML version which is easiest for working with on the screen, and a postscript version (also available in PDF) which is better for printing out.

Type ‘‘netscape’’ to run a web browser on the ECF unix machines. Web pages are identified by a ‘‘URL’’, which is a string consisting of ‘‘http://’’, an internet host name, another slash, and then a path name on that host. To look at the CSC 180 labs, enter the URL:

```
http://www.dgp.toronto.edu/~ajr/180/lab/
```

You can ‘‘bookmark’’ this URL so that you can go back to it quickly in future. In unix, the web browser ‘‘bookmarks’’ are stored separately for each user.

13. ‘‘Usenet news’’ is a facility somewhat similar to e-mail except that the messages are not personal; all messages which anyone posts are available for all to read. You can read the ‘‘newsgroup’’ for this course by typing:

```
rn -q ut.ecf.csc180
```

Do not post a test message. Wait until you have something to say, then post that.

The ut.ecf.csc180 newsgroup does not have any official status and we may or may not have time to read it or to respond to messages there. But it can be a useful resource for students to help each other, and we will read it some of the time.

14. When you are done with the computer, don’t forget to ‘‘log out’’. If you don’t log out, then whoever next uses the computer can impersonate you and modify your files, etc. On the other hand, if you come upon a computer on which someone has forgot to log out, you must log them out and log in as yourself.