COMP-232 Unix/C Tutorial
LAB 3 –sftp / ssh / putty

John Dempsey
COMP-232: Programming Languages
California State University, Channel Islands
September 10, 2025
Hard Due Date: September 19, 2025

Your user name **<username>** is your first name in lower case, e.g., john. Your password is **comp232**. If there are two students with the same first name, e.g., bryan, your last name is your user name in lower case.

Once logged on, change your password. To do this, type:

% **passwd**

Enter your new password twice.

Make sure you remember your user id and password for future use!

Run the following commands:

% **ls** ← Nothing is there. This is a new account.

% **whoami** ← Who am I logged in as.

% **grep <username> /etc/passwd** ← Did I spell your name correctly?

Let's understand the /etc/passwd file. Here are the entries for Bryan's entry. Each field is delimited by the colon.

bryan:x:1167:1166:Bryan Zhao,,,:/home/bryan:/bin/bash

bryan – user name
1167 – user id
1166 – group id as found in the /etc/group file
Bryan Zhao – Full name
/home/bryan – Bryan's home directory
/bin/bash – Bryan's default shell used when logging in

% **grep <username> /etc/group** ← I assigned you to your own group.

Again, let's look at Bryan's entry in /etc/group:

bryan:x:1166:

bryan – group name (yes, I know. This isn't usual. Usually your group would be assigned to a common group like staff, representing the staff group).

* – group password. If empty, no password exists.

1166 – group id

user list – You can add users to be in your group.

% **touch test** ← Create an empty file

% **ls -l test** ← Let's check who owns the file and what group owns the file.

Now start up an Ubuntu terminal window on your laptop.

Change to the directory where you have your 1.txt, 2.txt, 3.txt, and 4.txt; myfor.c, mywhile.c, myswitch.c, and myif.c; and, age.c source code files.

% **sftp <username>@comp232.com** ← Or you can use: **sftp samson@143.198.238.179**

sftp> **pwd** ← Print working directory on the remote system

sftp> **mkdir LAB1** ← Use capital letters for directories as they are listed first.

sftp> **cd LAB1** ← Change to LAB1 directory

sftp> **put lab1.txt**

sftp> **mkdir TASK1 TASK2 TASK3** ← Create TASK subdirectories.

sftp> **pwd** ← Print working directory on the remote system

sftp> **lls** ← List files on your system.

sftp> **ls** ← List files on comp232.com

sftp> **cd TASK1**

sftp> **put 1.txt**

sftp> **put 2.txt**

sftp> **put 3.txt**

sftp> **put 4.txt**

sftp> **ls** ← Make sure 1.txt, 2.txt, 3.txt, and 4.txt are there.

sftp> **cd ../TASK2** ← Now cd to TASK2 directory on comp232.com

sftp> **put myfor.c**

sftp> **put mywhile.c**

sftp> **put myswitch.c**

sftp> **put myif.c**

sftp> **cd ../TASK3**

sftp> **put age.c**

sftp> **cd ..**

sftp> **ls -lr** ← List all files and TASK directories

sftp> **quit** ← All done.

Now your files for LAB 1, TASKS 1-3, are on comp232.com.

Let's make sure they compile cleanly using the gcc compiler on the remote system (comp232.com system, that is).

You have two options to log into comp232.com: use ssh (secure shell) or use putty. To use ssh, type:

% **ssh <username>@comp232.com** ← Use ssh to log into comp232.com.

% **cd** ← Change to your home directory on comp232.com

% **cd LAB1** ← Change to where your source code files and output files are.

% **ls -lr** ← Let's see what's here.

Make sure you can compile your programs with no errors or warnings using:

% **cd TASK2**

% **gcc myfor.c -o myfor**

Now run each program on comp232.com to make sure they work.

% **ls -l myfor** ← Note it's executable file.

% **myfor**

But wait, when I run it, you might see:

myfor: command not found

Why is that? And let's correct this now.

% **echo $PATH** ← This will show you where executables are searched for in the order of the PATH. And you notice that your current directory isn't listed.

% **./myfor** ← This forces you to run the ratio command found in your current directory.

OR

% **PATH=$PATH:.** ← This will add your current directory you're in to the PATH, but only for this current session. Don't forget the dot after the semicolon.

OR

% **cd** ← Change to your home directory (as defined in /etc/passwd)

% **ls -l .profile** ← No .profile here.

% **vi .profile** ← New file, so simply add the following one line:

PATH=$PATH:.

% **. .profile** ← You can log out and log back in, or simply run your .profile to update PATH.

% **echo $PATH** ← The dot is at the end of the path. You're good every time you login now.

Now run each of your programs to make sure they work as expected.

Now that you updated your PATH, make sure you can compile and run your TASK2 and TASK3 programs on comp232. by running:

% gcc mywhile.c -o mywhile ← Do this for all programs in TASK2

% **mywhile**

Do the same for myif.c and myswitch.c.

% **cd ../TASK3**

% **gcc age.c -o age**

% **age**

% cd ..

% **ls** ← **When done, LAB1 directory should contain three directories named:**

TASK1 TASK2 TASK3

Putty

Installing Putty is optional, but it is a very useful tool for logging into remote systems used at work.

First, check to see if you have putty installed on your system.

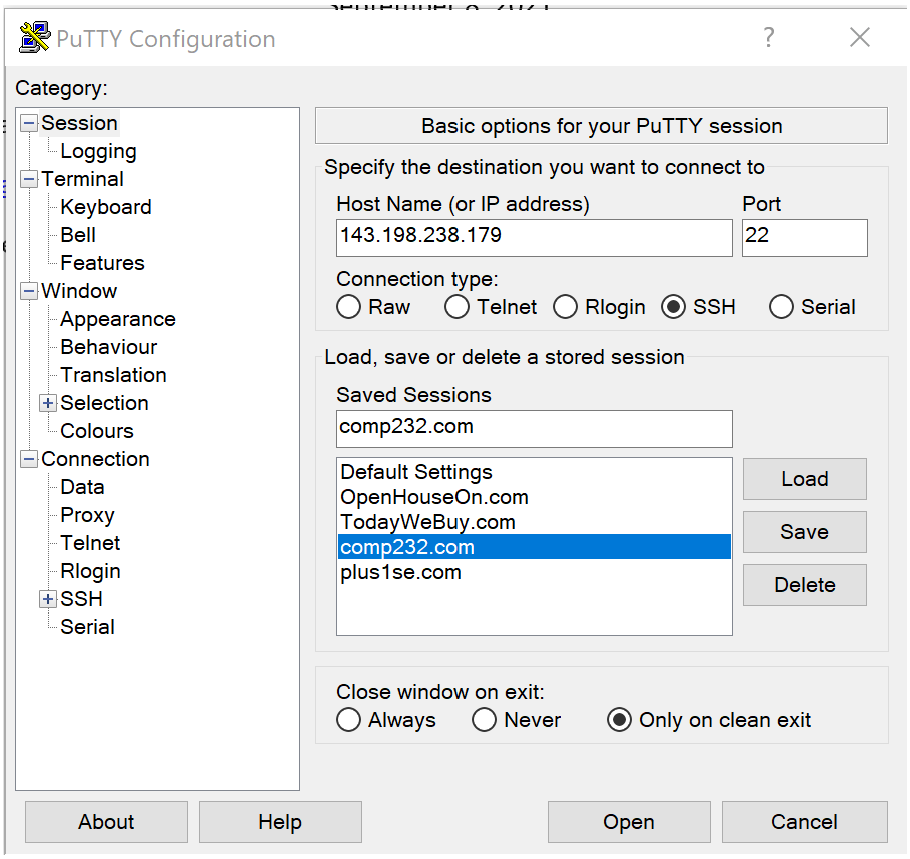
If not, the official download site for Putty for Window 10/11 is:

PuTTY: a free SSH and Telnet client (greenend.org.uk)

Mac users can download putty from:

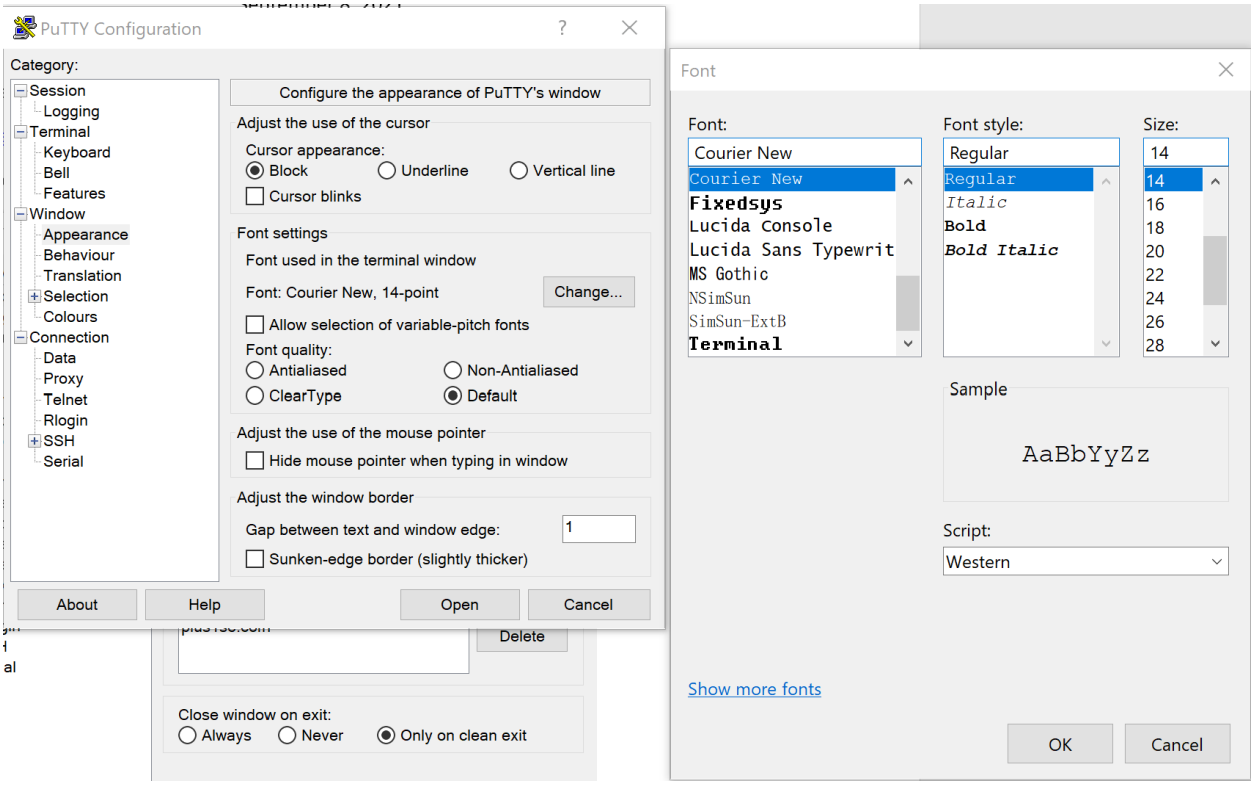
Download Putty (0.76) for Windows, Linux and Mac - Install SSH in PuTTY (puttygen.com)

Start Putty. Enter 143.198.238.179 for Host Name/IP address and comp232.com for Saved Sessions.



A common Putty change is to pick a font and/or font size rather than the default. To do this, click on **Appearance -> Change**. The Font window will appear. Select a font size larger than the default. Click OK.

But to save your changes, you need to select **Session** again, and then hit the **Save** button.



To log into comp232.com (after clicking on Save), click on the **Open** button.

A Putty window will appear.