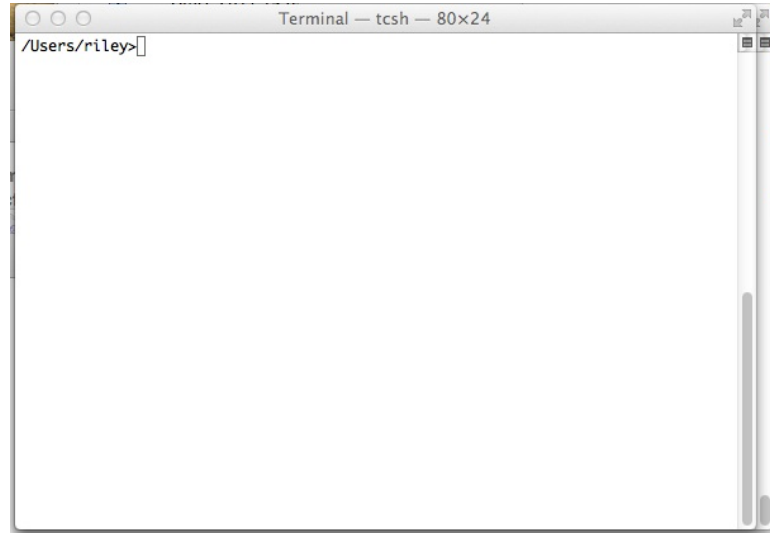


## Using SSH to Connect to deeptthought

*ECE2036, Spring 2016*

*George Riley*

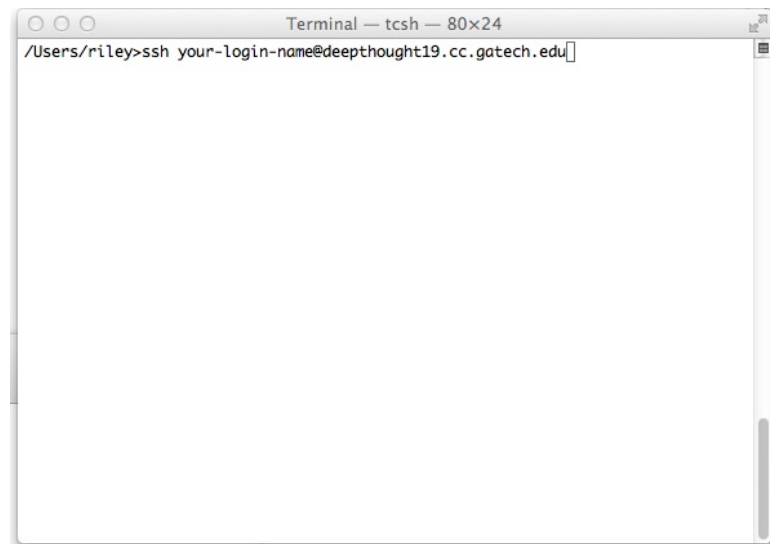
1. If you use Windows and have not yet installed *XWin32* you should go to <http://software.oit.gatech.edu>, select your affiliation as student and the software type is Microsoft Windows Desktop. Then select and download X-Win32 2014 (with SSH).
2. Bring up a terminal window on your laptop. If you have a Mac or Linux it is simply the Terminal application. For windows use the XWin32 downloaded above. Below is a screen shot of the terminal window on Mac.

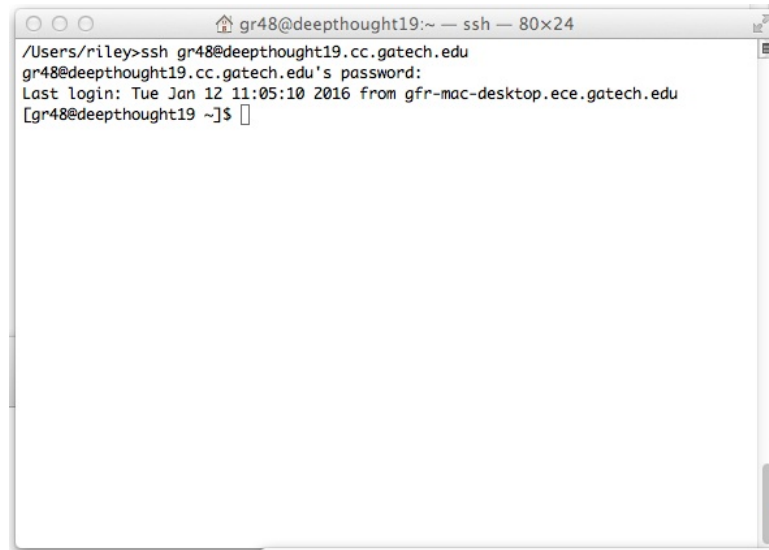


Notice the prompt showing the current working directory (`/Users/riley` in my case. Yours will of course be different.

3. Use the `ssh` command to “connect” to a different computer, `deeptthought19.cc.gatech.edu` in this case.

Below are screen shots before and after entering the final “return” key and after entering your log in password.

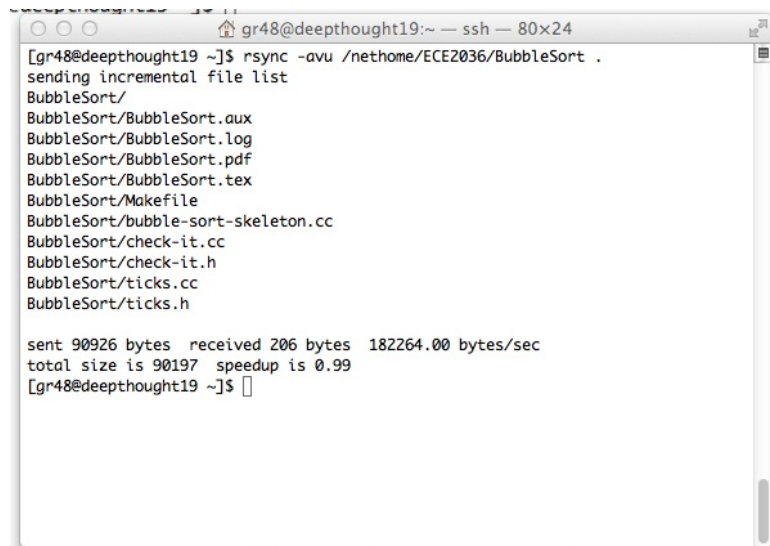


A terminal window titled "gr48@deeptought19:~ — ssh — 80x24". The prompt is "/Users/riley>". The user enters "ssh gr48@deeptought19.cc.gatech.edu". The terminal shows the login process: "gr48@deeptought19.cc.gatech.edu's password:", "Last login: Tue Jan 12 11:05:10 2016 from gfr-mac-desktop.ece.gatech.edu", and the prompt "[gr48@deeptought19 ~]\$".

```
/Users/riley>ssh gr48@deeptought19.cc.gatech.edu
gr48@deeptought19.cc.gatech.edu's password:
Last login: Tue Jan 12 11:05:10 2016 from gfr-mac-desktop.ece.gatech.edu
[gr48@deeptought19 ~]$
```

Note that the “Prompt” is slightly different on deeptought in that it prints the login name followed by the local machine name (deeptought19 in this case).

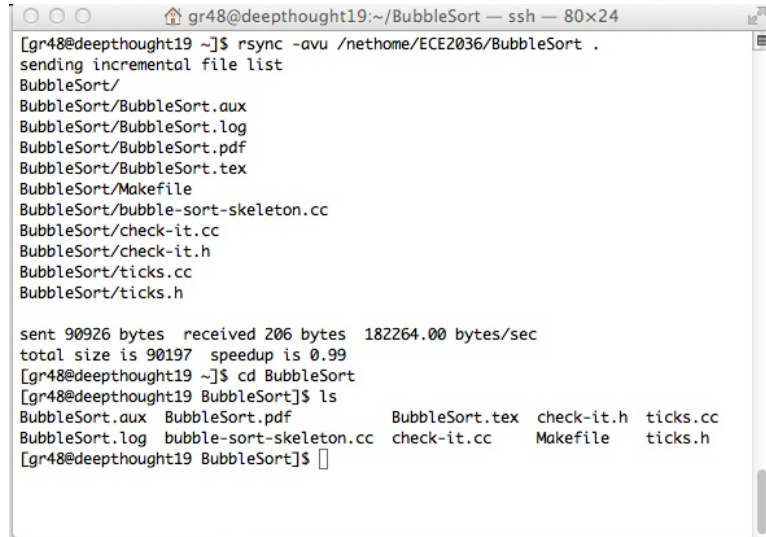
- Below is an example of how to download the *Skeleton* code that is usually provided for the C and C++ programming assignments.

A terminal window titled "gr48@deeptought19:~ — ssh — 80x24". The prompt is "[gr48@deeptought19 ~]\$". The user enters "rsync -avu /nethome/ECE2036/BubbleSort .". The terminal shows the output: "sending incremental file list", a list of files, and transfer statistics. The prompt is "[gr48@deeptought19 ~]\$".

```
[gr48@deeptought19 ~]$ rsync -avu /nethome/ECE2036/BubbleSort .
sending incremental file list
BubbleSort/
BubbleSort/BubbleSort.aux
BubbleSort/BubbleSort.log
BubbleSort/BubbleSort.pdf
BubbleSort/BubbleSort.tex
BubbleSort/Makefile
BubbleSort/bubble-sort-skeleton.cc
BubbleSort/check-it.cc
BubbleSort/check-it.h
BubbleSort/ticks.cc
BubbleSort/ticks.h

sent 90926 bytes  received 206 bytes  182264.00 bytes/sec
total size is 90197  speedup is 0.99
[gr48@deeptought19 ~]$
```

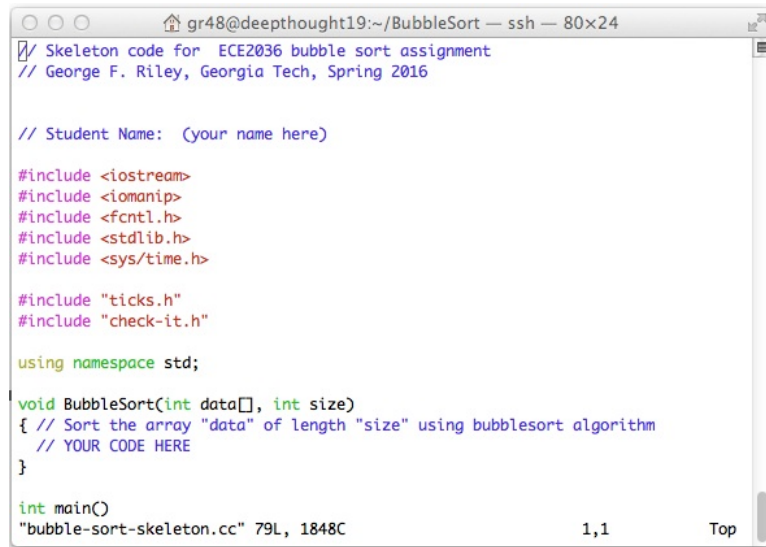
- Finally is an example of the output after changing your working directory using `cd` and then a `ls` (list files) command.



```
gr48@deepthought19:~/BubbleSort — ssh — 80x24
[gr48@deepthought19 ~]$ rsync -avu /nethome/ECE2036/BubbleSort .
sending incremental file list
BubbleSort/
BubbleSort/BubbleSort.aux
BubbleSort/BubbleSort.log
BubbleSort/BubbleSort.pdf
BubbleSort/BubbleSort.tex
BubbleSort/Makefile
BubbleSort/bubble-sort-skeleton.cc
BubbleSort/check-it.cc
BubbleSort/check-it.h
BubbleSort/ticks.cc
BubbleSort/ticks.h

sent 90926 bytes  received 206 bytes  182264.00 bytes/sec
total size is 90197  speedup is 0.99
[gr48@deepthought19 ~]$ cd BubbleSort
[gr48@deepthought19 BubbleSort]$ ls
BubbleSort.aux  BubbleSort.pdf  BubbleSort.tex  check-it.h  ticks.cc
BubbleSort.log  bubble-sort-skeleton.cc  check-it.cc  Makefile  ticks.h
[gr48@deepthought19 BubbleSort]$
```

6. Next is a screen shot of using the `vi` text editor. In this case we have specified to run the `vi` program on the `bubble-sort.cc` file. There are many good on-line resources about how to use `vi`.



```
gr48@deepthought19:~/BubbleSort — ssh — 80x24
// Skeleton code for ECE2036 bubble sort assignment
// George F. Riley, Georgia Tech, Spring 2016

// Student Name: (your name here)

#include <iostream>
#include <iomanip>
#include <fcntl.h>
#include <stdlib.h>
#include <sys/time.h>

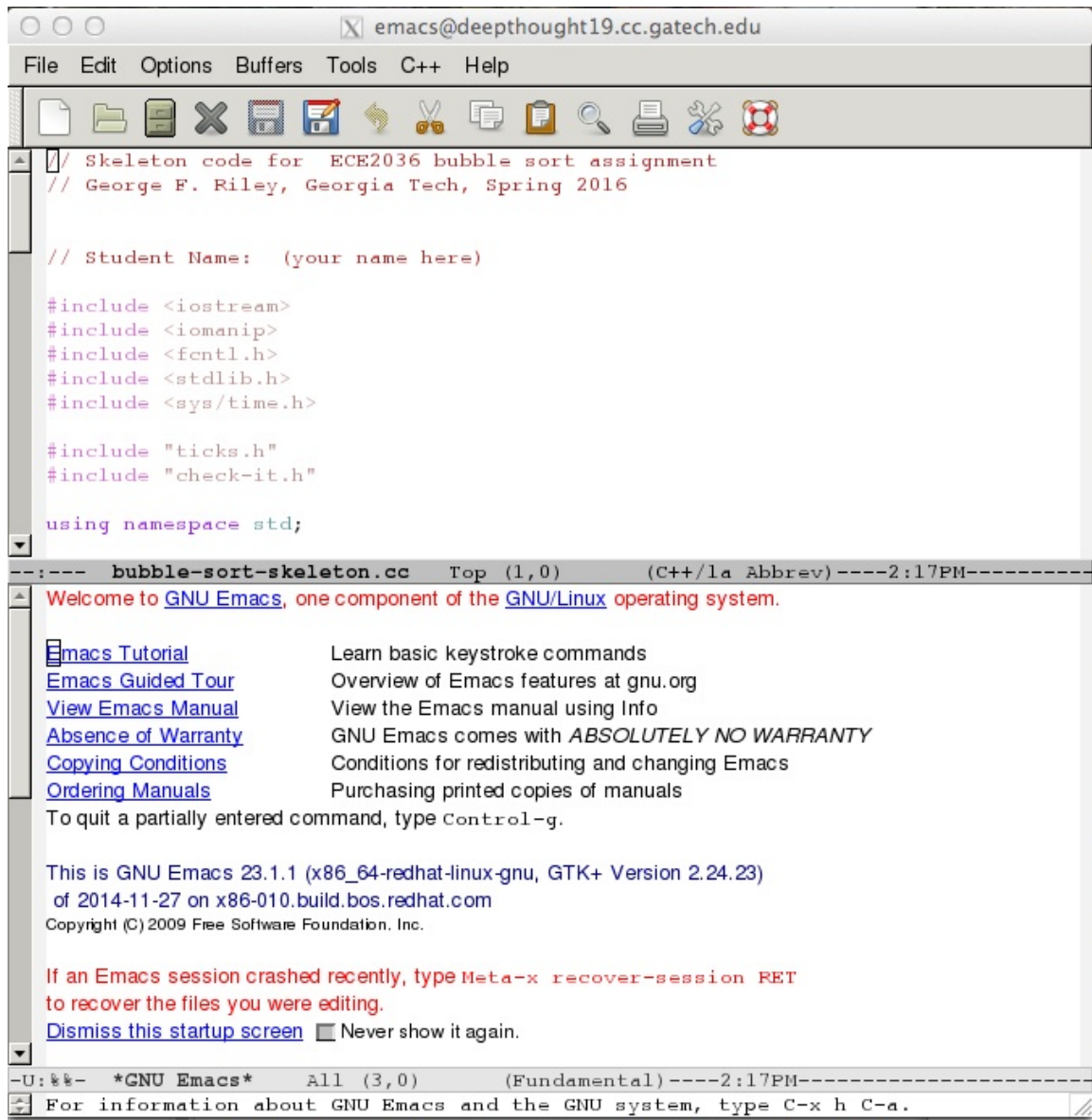
#include "ticks.h"
#include "check-it.h"

using namespace std;

void BubbleSort(int data[], int size)
{ // Sort the array "data" of length "size" using bubblesort algorithm
  // YOUR CODE HERE
}

int main()
"bubble-sort-skeleton.cc" 79L, 1848C                                1,1                                Top
```

7. Finally a screenshot of using `emacs` to edit your `bubble-sort.cc` file. Notice that `emacs` opens a second and larger window with the text of the program. This will only happen if you have `X11` installed on your laptop. For linux systems it is included with the distribution, but with Mac OSX there is a separate install process for `X11` called `XQuartz`. For windows platforms the `XWin32` discussed above should be sufficient.



8. There are a few linux commands that are needed to successfully complete the assignments. There are dozens of on-line resources for learning linux shell commands. A few of the useful commands are given below.
- `ls` - List all files in the current directory excepting *hidden* files.
  - `ls -la` - List in long format all files in the current directory including *hidden* files.
  - `cd SomeDirectoryName` - Change your working directory to some other directory. If the `SomeDirectoryName` starts with a forward slash, the directory is relative to the system *root* directory. You would rarely do this.
  - `cd` - Change your working directory to your *home* directory.
  - `cd ..` - Change your working directory to the *parent* of the current directory.
  - `make program-name` - Run the compiler and linker to create an executable binary `program-name` using the source code `program-name.cc`
  - `g++ -o program-name -g -Wall program-name.cc` Run the C++ compiler, compiling `program-name.cc` and writing the executable binaries on file `program -name`