More Useful Unix Commands

Here are a few more commands, especially useful when you're using a public machine, and trying to figure out where things are and what is installed:

- which: Will tell which version of the program it's using by default (introduced last lab). For example,

```
which g++
```

will show the location of the g++ program.

- man: This gives the built-in help pages for the current system. It can be quite useful if you don't remember all the options for a Unix command. For example,

```
man sort
```

will print out information about the command line sort.

- w: This command was designed for multi-user machines and tells you who is logged in and what they are doing. While you are the only one on your laptop, this command is still useful in that summarizes all the activity on the machine in a short table as well as how long the system has been running ('uptime'). So, if the machine seems sluggish or you're wondering how much else is going on, it will give a concise summary.
- more: Allow you to look at output or a file, one screenful at a time. This is incredibly useful when you are trying to read information that scrolls off the screen. Here's a sample of looking at all programs (from /usr/bin-- a common place to keep them) :

```
ls /usr/bin | more
```

The pipe ('|') is explained in Lab 12. Say we're trying to figure out what GNU C/C++ compilers are on the machine, and as a start want to see all the possible programs that contain "cc" (anywhere in their name):

```
ls /usr/bin | grep cc | sort | more
```

We don't see g++ in the list, so, let's also look for all that start with g:

```
ls /usr/bin | grep ^g | sort | more
```

The grep ^g asks for all names that start with "g". grep is the pattern matching command mentioned briefly in Lab 9 (for an overview see the grep wiki page).