

# MAT 101 — Introduction to Computer Programming (C++), Spring 2005

Handout #1 - January 20, 2005

## Logging into *albert*:

1. Boot with Knoppix (the linux boot disc)
2. Double click on the *monitor* icon to open a terminal
3. type "ssh -X *username*@albert.washcoll.edu" where *username* is the address you use for email.
4. type "emacs" to open the text editor

## Using *emacs*:

Key-stroke	Effect
alt-	Meta key. M-x means press alt-x
ctrl-x ctrl-f	Open a file
ctrl-x ctrl-s	Save the current file
ctrl-x ctrl-c	Close emacs
ctrl-d	delete
ctrl-a	beginning of line
ctrl-e	end of line
ctrl-b	back one character
ctrl-f	forward one character
ctrl-v	forward one page
M-v	back one page
M-x global-font-lock-mode	Turn on C++ colors
M-x global-set-key RET F4 RET compile RET	define F4 to be the compile key

## A Simple Makefile

Type 'emacs Makefile' to open a file called Makefile. This should be in the directory in which you plan to be working on your project. Here is an example of a simple Makefile:

```
hello: hello.cpp
    g++ hello.cpp -o hello
```

In this simple example, we are creating an *executable* named hello. An executable is a file that can actually be run. It can do anything we program it to. For quite awhile, this will be very simple things as we learn how to program. In this example, The *source* file is called hello.cpp. A source file is a file that has program code in it. Its something written in text that humans can read, write, change and understand. The object is to change source code into an executable. In this example, the executable file 'hello' is *dependent* on the source file hello.cpp. For now, create your Makefiles like this. In the future you may learn how to create more intelligent make files. In this file, if hello.cpp is modified after the hello file was created, then the next line (the g++ command) is executed. Note that the second line begins with a tab and not spaces. g++ is a compiler and that line *compiles* the source file hello.cpp into an executable file called hello, as we desired. If we code our source files (hello.cpp) correctly, we will now have a file called hello that can be executed. Let's write a simple hello.cpp program and compile it, while we're in class. Your hello.cpp file should look like this:

```
#include<iostream> // this line allows us to do output statements
using namespace std; //tells us what kind of output statements to use

int main () // this begins our statements that will be executed
{
cout << "Hello world" << endl; // an output statement
}
```

Next time, we'll set up colors and F4 automatically when you start up emacs. This can be done by placing a couple commands in a file called '.emacs' in your base directory. Your .emacs file should look like:

```
(global-font-lock-mode t)
(global-set-key [f4] 'compile)
```

There are many emacs tutorials. Try googling for another tutorial or the answer if you can't find it here or in the tutorials listed below.

1. [http://www.delorie.com/gnu/docs/emacs/emacs\\_toc.html](http://www.delorie.com/gnu/docs/emacs/emacs_toc.html)
2. <http://www.lib.uchicago.edu/keith/td-course/emacs-tutorial.html>
3. [http://vertigo.hsrl.rutgers.edu/ug/emacs\\_tutorial.html](http://vertigo.hsrl.rutgers.edu/ug/emacs_tutorial.html)