APPENDIX B

UNIX

In this first lab you will learn how to create, compile, and run C++ programs using the UNIX operating system.

I. Pre-lab Assignment

1. Be sure to read the Lesson Set 1 Pre-lab Reading Assignment and then do the Pre-lab Writing Assignment before coming to class.
2. Make sure you have a login name and a working password to logon to a UNIX system. Your instructor, along with the UNIX administrator, will help you obtain an account and learn how to login.

II. The UNIX Operating System

UNIX is a very powerful operating system. However, for some users it is not quite as easy to learn as the Windows environment. UNIX requires some extra knowledge on the part of the user. This appendix should help you learn some of the basics of this operating system. Nevertheless, you will most likely need additional information from your instructor or other sources to be a successful UNIX user. This appendix is a terse, but in no way complete, introduction to UNIX.

The UNIX system organizes files in a hierarchical structure. The top of this structure is called the root directory identified by `(/)`. Major directories are located under the root directory. For example, `/home` indicates the user home directory and `/bin` is the binary directory for executables. A directory called `programs` under `/home` is identified by `/home/programs`. The *path* `/home/programs/program1.cpp` shows the location of the C++ program `program1.cpp` in the `/programs` directory.

III. The UNIX Reference Manual

There is a UNIX reference manual that can be accessed during your UNIX session as long as your system has the `man` program. To get information regarding a particular command type `man` at the shell prompt followed by the desired command. For example,

```
man mkdir
```

provides you information about the `mkdir` (make directory) command. A list of useful commands that you may wish to explore as follows:

`cd`	changes directory
`cp`	copy file
`find`	find files
`grep`	locate text within a file
`ls`	displays contents of current directory

`man`	UNIX reference manual
`mkdir`	makes new directory
`mv`	move and rename a file
`pwd`	display name of current directory with path
`rm`	remove (delete) files
`rmdir`	removes directory
`vi`	begin a `vi` (visual interpreter) editor session

Of course, there are many other useful UNIX commands. To learn more about these commands and other important information, visit the UNIX resources at your library and local bookstore.

IV. Editing with `vi`

You can create and edit files during a UNIX session using the `vi` (visual interpreter) editor. First choose the desired subdirectory for your file. For example, if you are in the `/home` directory, enter `cd programs`. To start the editor enter `vi` at the shell prompt followed by the name of your file:

```
vi program1.cpp
```

There are two modes for `vi`: the text-entry mode and the command mode. First select the text-entry mode by typing the character **`a`** for append. Now you are ready to enter text.

Type the following:

```
#include <iostream>
using namespace std;

int main()
{
     cout << "Welcome to the UNIX operating system" << endl;
     return 0;
}
```

You do not need to worry about what the code means at this point. Just make sure you can enter this using the `vi` editor. To exit the text-entry mode, press the `Escape` key. This will return you to the command mode. Editing commands can be entered from this mode.

A simple example of a `vi` editing command is the command to delete a character. If you want to delete a single character, move the cursor to the desired character and type **`x`**. To delete multiple successive characters, move the cursor to the first character and type the desired number followed by **`x`**. For example, **`9x`** deletes the next 9 characters. Other editing commands are the following:

`dw`	deletes a word
`dW`	deletes a word that contains punctuation
`ndw`	(or `ndW`) for deleting successive words
`i`	inserts new text (changes to text-entry mode)
`O`	opens a new line above (changes to text-entry mode)
`o`	opens a new line below (changes to text-entry mode)

For more information on these and other editing commands use the `man` command described above or some other resource on the `vi` editor.

To save the program you just entered, make sure you are in command mode and type **:w** and then press [ENTER]. Your work should now be saved in `program1.cpp`.

Finally, you will need to exit the `vi` editor before you can compile and run your program. To do this, type **:q** and then press [ENTER]. The shell prompt should now appear on your screen.

V. Compiling and Executing a C++ Program

Possible commands to compile your C++ program will depend on your particular system. Many UNIX systems have GNU C/C++ compile. In this case you may type

```
g++ program1.cpp
```

to compile your program. Assuming you typed everything correctly, there should be no errors and an executable will be created. A shell prompt will appear without any messages. Now type `ls` and then press [ENTER]. You should see a new file listed named `a.out`. To execute your program, type **a.out** and then press [ENTER]. The following should appear on your screen:

```
"Welcome to the UNIX operating system"
```

This indicates that you have executed your program. If `g++` does not successfully compile your program, see your instructor to find out the command necessary for your particular system.

Often, you will want to bring in an existing C++ program from a floppy disk or some other source. Your instructor will show you how to retrieve existing files from other locations.

VI. Logging Off

To log out upon completion of your UNIX session, type `logout` and then press [ENTER]. Make sure that you do not misspell `logout`—otherwise you may be leaving your files open to the next user!