CIS 200 Lab 1 Fall 2016

**READ the deliverables before starting the actual task so you know what you must turn in for each task. You can put all the screenshots / output in a Microsoft Word document and submit that document.**

To take a Screenshot of a **single** window, use **Alt + PrtScn** on the keyboard. To take a screenshot of the entire window, just press PrtScn.

# Task 1

For this task, you will use putty to connect to **login.umd.umich.edu**, specifying your username and your password when prompted.

1. Make a directory named cis200 using mkdir
2. Move into that directory using cd
3. Use the **pwd** command to print the working directory. This should be your home directory/cis200
4. Use **ls** command with the appropriate options (a and l) to list all files and folders in long format, including hidden files.

## Task 1 Deliverables

You should take a screenshot of the output of both the **pwd** and **ls** commands (with appropriate flags/options) and paste it into your Word document.

# Task 2

1. Create a basic “Hello World” program using Visual Studio on Windows, naming the source file **hello.cpp**.
2. Verify that it works (compiles and executes) through Visual Studio.
3. Then, use the winscp and transfer the **hello.cpp** file to your Unix account.
4. Compile the source file using **g++** and using the **-o** option to name the executable hello
5. Run the executable you just created (./hello)

## Task 2 Deliverables

Take a screenshot of the Unix commands you issued to compile and run the hello executable. The output from your program should be visible as well. Make sure that you paste the screenshot in the Word document you are going to turn in.

# Task 3

1. Create a new file from inside of your command line using the pico or vi command, naming it hello2.cpp
2. Write the appropriate code, using the pico editor, to print “Hello, Professor!” to the console
3. Save the file and exit
4. Compile the hello2.cpp file and create the executable with the name hello2
5. Execute the hello2 program

## Task 3 Deliverables

Take a screenshot of the Unix commands you issued to compile and run the hello executable. The output from your program should be visible as well. Make sure to paste the screenshot into the Word document you are going to turn in.

# Task 4

1. Use the **mv** (move) command to change the name of hello2 executable to hello\_prof.
2. Execute and run this program

## Task 4 Deliverables

Take a screenshot of the Unix commands you issued to change the name of the hello2 executable. Also, the screenshot should show the output from the hello2 program.

# Task 5

1. Create a subdirectory named **lab1assignment** under the cis200 directory you made earlier
2. Using the cp command, copy all the ***source files*** (.cpp) that you created (do not copy the executables)
3. Using the rm command, remove all the files from the top level of the cis200 directory, but leave the copies in tact in the lab1assignment directory

## Task 5 Deliverables

* Take a screenshot of the **cis200** directory with **pwd** and **ls** commands issued (to show you’re in the cis200 directory, and to show the contents of that directory)
  + This directory should only have the subdirectory in it (lab1assignment) and not the source and executable files (that you copied to lab1assignment, and then deleted from the cis200 directory)
* Take a screenshot of the newly created **lab1assignment** directory with **pwd** and **ls** commands issued
  + This directory should contain the source files only

## Some Useful Unix commands

|  |  |  |
| --- | --- | --- |
| Command | Description | Examples |
| echo param | Used to echo the parameter. If an environment variable, the contents will be displayed | echo $SHELL |
| mkdir dirName | Used to make a directory (folder) in which you can store files | mkdir lab1\_dir |
| cd dir | Used to change directory | cd lab1\_dir cd .. |
| pwd | Used to print the path of the current directory | pwd |
| ls | Used to print the contents of the current directory. Some Options:   * ls lists all non hidden * ls -a lists all * ls –l lists in long format * ls -al lists all in long format | ls -a  ls  ls -l  ls -al |
| cat filename | Used to display the contents of a text file without opening it | cat someFile.cpp |
| mv file1 file2 | Used to move *and/or* rename a file | mv myFile myOther mv myFile direct/ |
| cp file fileCopy | Used to copy a file | cp myFile myFileBackup |
| rm filename | Removes a file | rm someFile |
| rmdir dirName | Removes a directory (folder) | rmdir myDirectory |
| vi filename | Opens vi (or vi improved, known as vim) and creates a file | vi test.cpp |
| g++ filename.cpp | Compiles a C++ source file. If no output flag and name are specified, the executable is generated with the name **a.out** If you use the -o flag with a name, the executable will be generated with that name | g++ test.cpp g++ test.cpp -o test |

## Vi Editor

The vi editor runs in two modes

* Insert mode
* Command mode

When you first open up vi, you are in **command mode**. This mode allows you to navigate through the file and also to issue commands. In this mode, all keystrokes are considered commands, NOT text to be added to the file.

Some useful commands:

**a** append right after current position

**[Shift]+a**  append to end of line

**:w** Write file (save)

**:q** Quit vi

**:q!** Force quit (quit without saving)

**i Enter insert mode**

To enter insert mode from command mode, just hit the **i key.**

To return to command mode, hit the **Escape key.**