**CS365 Operating System Lab 1**

***I You need to know Unix***

Read three articles, Wikipedia is OK, about Unix (or it is unix or UNIX). The truth is that most (someone counted 80%) of software engineering jobs are Unix based, so learn something about Unix!!!

***II getting an environment***

1. ***Ubuntu – you have several options here***
2. Ubuntu is your only operating system.

Following the installation instruction.

1. Install Ubuntu with window.

http://www.ubuntu.com/download/desktop/install-ubuntu-with-windows

1. Install Ubuntu using VirtualBox

<http://www.psychocats.net/ubuntu/virtualbox>

1. Install Ubuntu using Docker
2. ***Cygwin***

Start at <https://www.cygwin.com/> and find your own resources to learn about it. Here are two <http://lifehacker.com/179514/geek-to-live--introduction-to-cygwin-part-i> and <http://lifehacker.com/179514/geek-to-live--introduction-to-cygwin-part-i>.

\*\*\* Make sure you install gcc-core, by default it is not installed. It is under the Devel Default category \*\*\*

1. ***Bash shell for Windows 10***

This is very new, meaning you should try this so you can put this on your resume. Following this link <http://www.windowscentral.com/how-install-bash-shell-command-line-windows-10>. You do need Windows 10.

***III HelloWorld***

Experience the world of c-programming as the early computer scientists did before GUI's and integrated development systems such as Visual Studio.

Use Google to find a sample HelloWorld program. The hardcore people use the vi pronounced vee eye. Here is an URL <http://www.cs.colostate.edu/helpdocs/vi.html>

gcc is the compiler that you use to compile your c code. Here is an URL

<http://ce.uml.edu/compile.htm>.

You should be able to find more compete instructions on the web. Make this one a very very simple one!!!

***IV fork() system calls – this is the key of this lab!***

Create a program that uses the fork() and getpid () system calls. There is a simple example in our text and a lot of them on the web. Find a simple one, get it to work, and then play with it so it does something cool!

The following URL <http://www.csl.mtu.edu/cs4411.ck/www/NOTES/process/fork/create.html>

can be very useful. Read and gain a good understanding before coding. Get the first example to work. Make the MAX\_COUNT to be 20. Change the MAX\_COUNT back to 200 or even higher for the second example.

***V Do something with this fork thing***

Let’s assume that there are two actions you would like to take. In that case, you can give one to the parent process and give the other to the child process. How to tell the difference between the parent and china process? This is the first thing you have to figure out.

Now the two actions, one action to find the largest element and the other actions could be to find the smallest element. So create an array of 1000 and populate it with random integers before the fork call. Do two actions, with the parent finding the largest element and the child processing finding the smallest element, and then each prints the result out. Even better, do that before the fork so you know the result is correct or not. Event better, do these recursively with one or two functions.

Here is the part you need two copies of the same array. Replace every element of the parent with the largest element and print out the total; replace every element of the child process with the smallest element and print out the total. Yes, I know you can calculate the total, but here we are asking you to do the actual replacing to prove a point.

***VI What to turn in and by when***

Answer which option in II you take and why.

Show your c HelloWorld program for III.

Turn in the screen shots of the results of IV and V, in a PDF file and drop the file in the Lab1 folder on Moodle before EOF 4/17/2017. I’ll test you on your code. Here I just need your code.

**Due 4/17/2017**