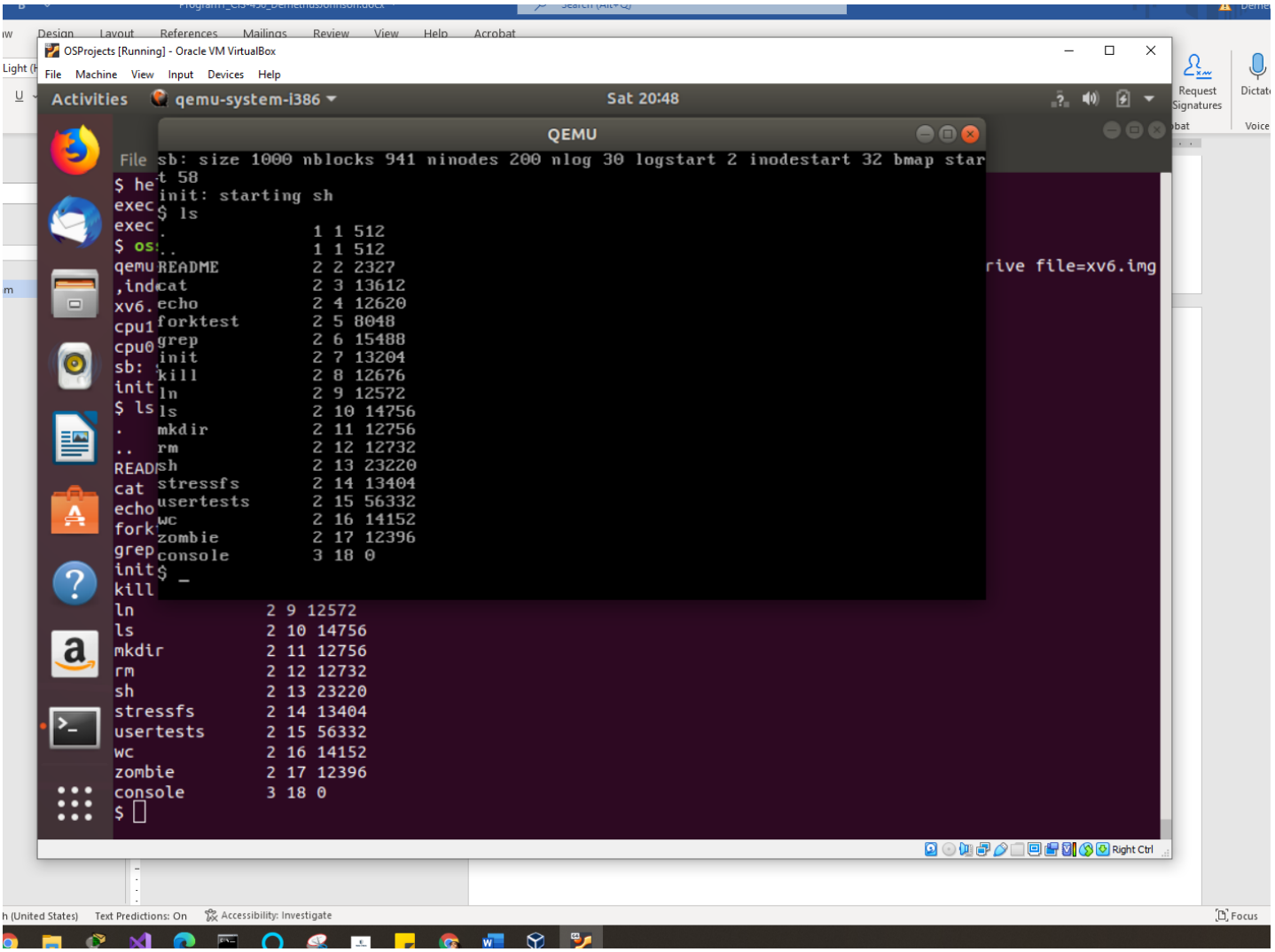
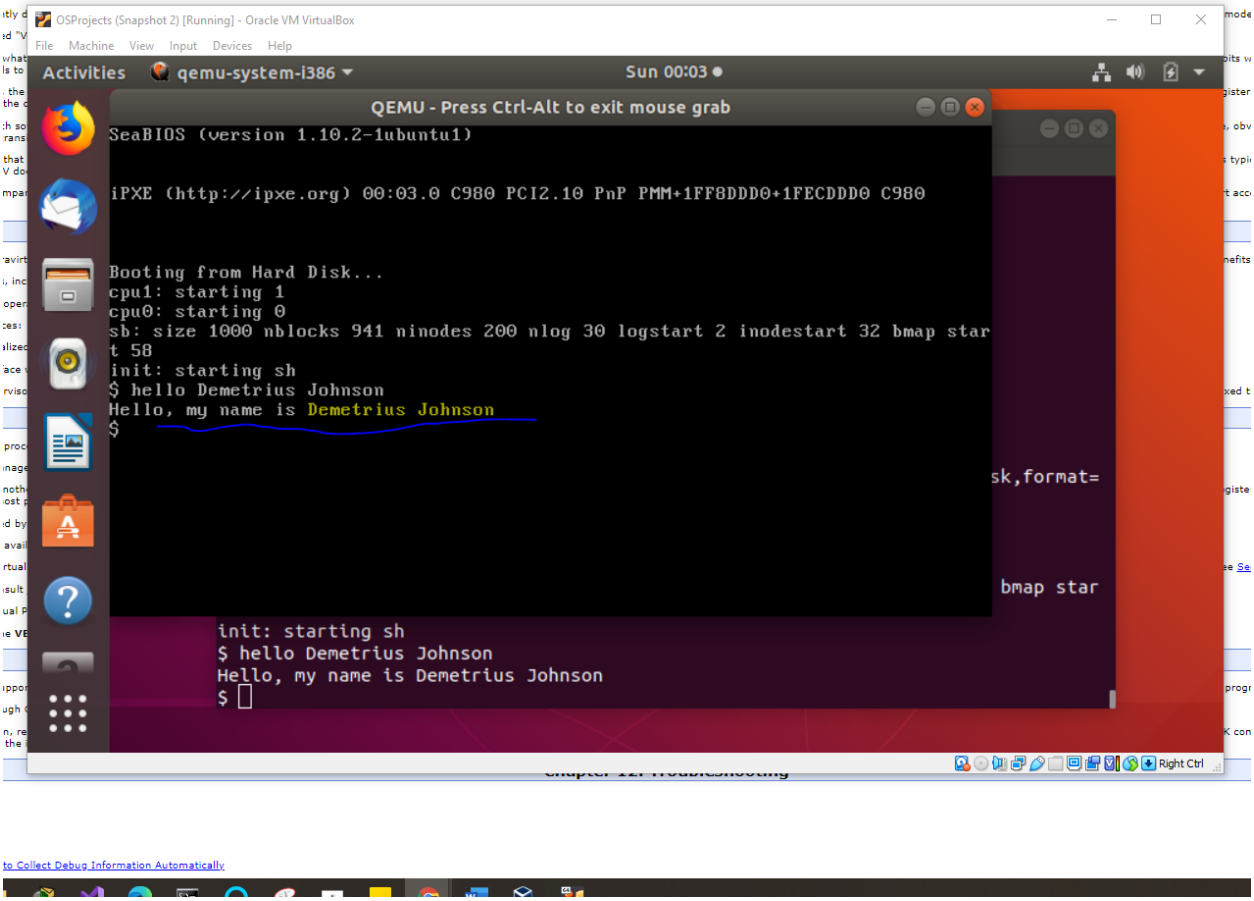
**Student Name: Demetrius Johnson**
**Course: CIS-450-002**
**Professor: Dr. Jinhua Guo**
**Date: January 29, 2022**
**Due Date: January 31, 2022**
# Project 1 Report

# Task 1 (screenshot 1)– Build and run xv6



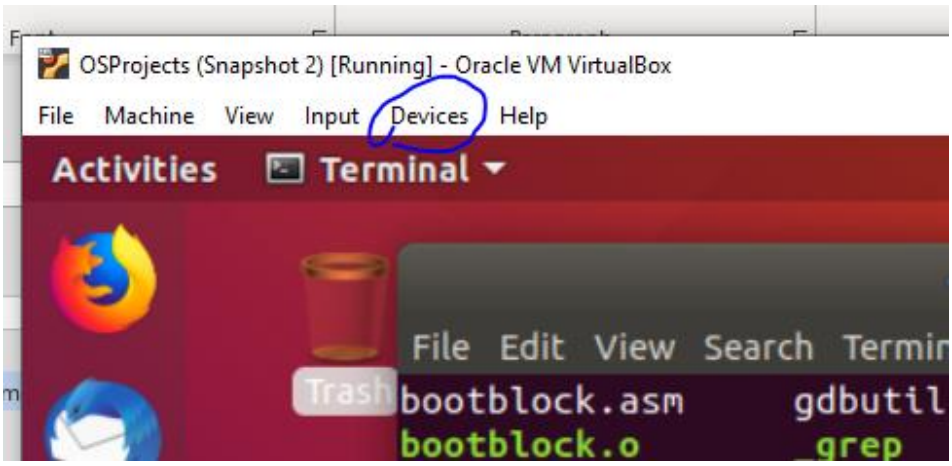

- Above, I have started my virtual machine through the VirtualBox hypervisor, made a copy of the xv6 source code so I can keep an untouched original copy, and then I compiled my copy of xv6, and then I compiled and ran the QEMU emulator in my virtual machine to run emulate the xv6 operating system batch files. Finally, I ran the ls command in the xv6 environment to show that the (emulated) operating system is functional.

# Task 2 (screenshot 2)– Adding a new user program




- Above, I have the output from running the hello program on the xv6 OS being emulated through QEMU. Also, I noticed that after issuing the “make” command on the Linux Ubuntu OS command line interface, multiple file types with the same name as my program was created. I noticed this after issuing the ls command inside of the xv6 (copied) directory. I will show a screenshot below:


    - Above, I noticed how _hello was created, along with hello.asm (which I know is an MASM assembly language file extension and I saw the assembly-level code when I opened the file), and many other files including an object file (hello.o). I used the "cat" command to look into the hello.d file, I assume that the .d extension stands for "directory", since it simply contained a file path. The rest of the file types with the name "hello" were unreadable, so I assume that must be the code used by the actual hardware of the computer to execute commands.
- Also, I realize that issuing the command "make qemu" compiles and runs a program called "qemu"; since I know that the program is an emulator, it must take the file path of my xv6 directory (where all of the xv6 source code is located) as essentially an input in order to simulate running the xv6 OS.
- Lastly, my Ubuntu VM actually crashed when I tried to install the updated Guest Additions, but I predicted this may have happened because I tried to do it after already running QEMU and then trying to update the Ubuntu OS Guest Additions to the latest version. The solution was to simply delete the VM and readd it just (essentially start from the beginning of this Project 1). Then, as soon as I ran the VM and the OS booted and I logged in, I immediately updated the Guest Additions and **rebooted** the VM (selecting the "UBUNTU" in the unusual 'bios-like' screen that came up before it would finish the reboot). Then I followed the instructions you gave in the video to install the VIM package. I then used the VIM editor to do the second part of this project. I had to do quite a bit of reading of the VirtualBox manual so I could do the installation and learn how to use it and take advantage of its capabilities, as well as get an idea of what kind of software it is (virtualization software – also known as a hypervisor or virtual machine manager) and how it works. I did the same for learning how to use the VIM editor.
    - Note, if the CD drive is not showing up on desktop, go to "Devices" in the window of your virtual machine and then at the very bottom it will say "Insert Guest Additions CD image". After you click that you click the "run program" button inside of the window that appears. See screenshot below:

- 
OSProjects (Snapshot 2) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Terminal
File Edit View Search Termir
Trash
bootblock.asm
gdbutil
bootblock.o
grep