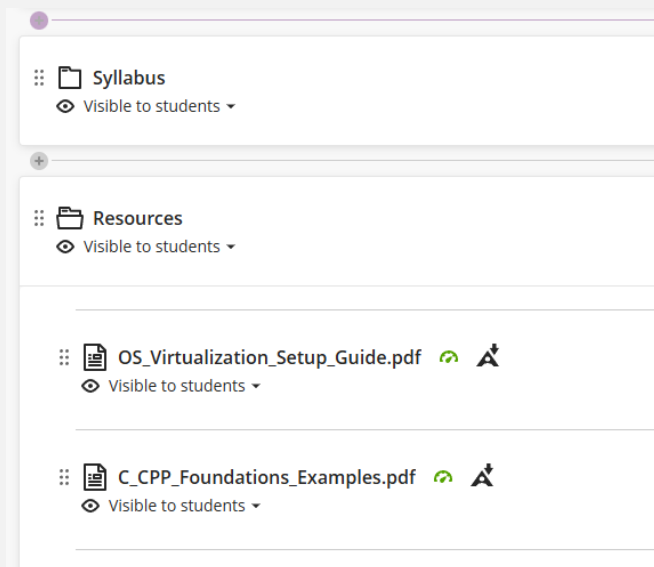

COSC 439: Operating Systems

Assignment#1

Objective:

This assignment will guide you through the process of setting up a virtual machine (VM), installing an operating system, configuring a C/C++ programming environment, and running a simple C/C++ program.

Note: Please check the **Resources tab on Blackboard** before starting. It contains necessary information and guidance regarding virtualization software and C/C++ programming.



Instructions:

1. **Install Virtualization Software:** If you don't already have one installed, choose the appropriate virtualization software for your computer:
 - a. **Windows or Intel-based Mac users:** Install **VirtualBox** or **VMware Workstation/Fusion**.
 - b. **Apple Silicon Mac (M1, M2, M3) users:** Install **UTM**, since VirtualBox and VMware are not fully supported.Ensure that the software is compatible with your operating system.
2. **Choose an Operating System:**
 - a. If your host is Windows, install Ubuntu (Linux).
 - b. If your host is Intel-based macOS, install Windows or Ubuntu.
 - c. If your host is Apple Silicon macOS (M1/M2/M3), use UTM and install the ARM-compatible version of Ubuntu or Windows (e.g., Windows 11 ARM). In general, if your primary laptop/desktop uses an ARM-based processor, you should install the ARM

version of the guest OS. Otherwise, the system may fail to run properly or may perform very poorly due to emulation.

The main goal is to gain experience with an operating system that is different from your primary host OS.

3. Set Up a Virtual Machine:

- a. Using your chosen virtualization software (VirtualBox, VMware, or UTM), create a new virtual machine and install the OS using the provided .iso file - [link](#)

4. Explore the Installed Operating System:

- a. **For Ubuntu:** Run the following commands in the terminal with the appropriate parameters (if any). Record the output and provide an explanation of what each command does:
 - i. ls
 - ii. pwd
 - iii. cd
 - iv. top
 - v. grep
- b. **For Windows:** Run the following commands in the Command Prompt with the appropriate parameters (if any). Record the output and provide an explanation of what each command does:
 - i. ipconfig
 - ii. ping
 - iii. dir
 - iv. tasklist
 - v. sfc /scannow

5. Environment Setup:

- a. Ensure that GCC or G++ is installed. Ubuntu usually comes with these pre-installed but double-check to confirm.
- b. Set up your preferred C/C++ Integrated Development Environment (IDE). Options include Code::Blocks, Visual Studio, CLion, or any other IDE of your choice. Linux users may also choose to use an IDE or continue using the editor and terminal to write and compile programs.

6. Run a Simple C/C++ Code:

- Write a basic C/C++ program: the program should read at least 5 numbers into an array, then output both the maximum and minimum values.
- Compile and run it on your VM.
- Ensure that the program executes without errors.

Submission Guidelines:

- Screenshots of the newly installed VM machine.
- Screenshots and explanations of the commands you ran in Step 4.
- Screenshots of the C++ code and its output on the newly installed VM machine.
- Combine all screenshots into a single PDF file and submit it.

- **Late submissions** will incur a maximum penalty of **25% of the marks**, so make sure to submit your assignment on time.

Note: If you already have multiple workstations running different operating systems (Windows, Linux, or macOS), you may use them for this assignment. The goal is to ensure you understand virtualization, using different OSs, how to install them using virtualization tools, how to configure a C/C++ environment in these OSs, and how to run simple C/C++ code. This foundational knowledge will be beneficial for your upcoming assignments.

If you encounter any issues that aren't addressed in the resources, I encourage you to search for the problem online first. Solving it yourself will deepen your understanding. However, if you are still stuck, feel free to email me or catch me after class. I am here to help!