

Operating Systems

CSCI 5806

Spring Semester 2025 — CRN 22968

Term Project — Overview

Due date: Friday, April 25, 2025

Goal

Design and implement a set of one or more programs that copy files into and out of a VirtualBox VDI file containing a Linux ext2 filesystem.

Details

The Linux ext2 filesystem is a basic implementation of the UNIX SysV filesystem with no journaling (ext3 and ext4 filesystems add journaling capabilities).

Your software will work with an ext2 filesystem contained within a VirtualBox VDI file; each of your programs should take the name of the VDI file as a command-line parameter.

Grading

For a passing grade (80%), your program must be able to do the following:

- Copy a specified file from the VDI file to the host system without corrupting the VDI filesystem
- Display all files and directories in the VDI filesystem

For an A grade (95%), your software must also be able to write a given file into the VDI filesystem. The filesystem should be readable afterward.

For full credit, your program must be able to write cleanly; that is, writing into the VDI filesystem should not cause issues with **fsck**.

For 5% extra credit, your software's file listing should provide the same information as a "long list" in UNIX (i.e., **ls -l**). You do not need to convert the UID or GID to a user name.

For an additional 5% extra credit, make your software work with dynamic VDI files, where disk pages aren't necessarily in sequential order.

Rules

Everyone must abide by these rules:

- You may work in pairs, or alone. Each person *must* send me an email indicating their partner, if any.
- Your source code must compile and run on Ubuntu 18.04. This largely precludes using Visual Studio.
- You may use any language, subject to the rules stated above.
- You may use an IDE, as long as I am able to compile the code without it.
- When submitting the project, send one zip file with the entire project folder.
- I only need one submission per pair. However, submissions from a group must identify both people.

Guidelines, Tips and Warnings

- I *very, very strongly* suggest using C or C++. Ignore this at your peril. You have been warned.
- I have broken the project down into steps to help you. Skip steps at your peril.
- I will be providing examples of what you should be seeing at the end of each step of the process.
- Each step should take no more than 1–2 weeks to complete. If you delay, or you take too long to complete a step, you will likely not finish the project.

What to turn in

Turn in all of your source code and any necessary auxiliary files in a zip file.