

# CIS 415 Operating Systems

## Project 1 Report Collection

Submitted to:

Prof. Allen Malony

Author:

*Elias Faris*

*efaris*

*951647865*

# Report

## Introduction

In our first project, we basically mimicked the Linux terminal using only system calls that take in user input. Also, there was file mode which reads a file and runs the commands in the file. These both needed a lot of error handling, and I will talk about this in my introduction. In interactive mode the user is prompted with, “>>> ”, here they would write a Linux command of the following, ls – which takes no parameters and displays what is in the user current directory. The next command I implemented was cd which takes a parameter and “goes into the parameter” which is a directory. I then implemented pwd which doesn’t take any parameters and shows what directory you are currently in. The next commands I tried to mimic were rm which removes a file and that takes one parameter. Then the next command was cat which reads and writes the file to stdout which also takes 1 parameter. Then the two commands that got me to work the hardest were cp and mv. They both take two parameters, a source and a destination.

## Background

First, I started off with the main file, to handle the user input and errors. This then set me up to work on our command file which held all the commands we implemented. I made two helper functions one that is called interactiveMode that finds out if we are in interactive mode and another function that helps us find out if we are in file mode that is called fileMode. Also, I made a validate function to return 0 or 1 to see if the command is valid or not. If not then it would print an error message explaining what’s wrong. Then I copied and pasted my Lab 1 functions to help find the tokens. Then I wrote all the commands in my command.c file to execute when they are called. I also had to look at the man page for Linux a lot and some examples that I found helpful. What else I found helpful that I think all courses should do for cis is, having lab be a small part of the homework. This way everyone will work on their homework before it is due and will have some confidence working on the main project. I also enjoyed the interactive mode of the project more than the file mode.

## Implementation

The problems I had on this project were small mistakes I got from not coding in C in over 4 months. This led me to passing in the wrong arguments and not dereferencing correctly. This caused me to segfault when I ran file mode and had problems printing ls; pwd. It thought that it was ls; ls instead of ls; pwd. I got it working but those bugs took the majority of my time and caused me to turn it in late.

## **Performance Results and Discussion**

When I was testing everything it all worked correctly, but when I run valgrind I am not sure why it prints out the address of my functions. When I run valgrind on file mode, I am missing a free spot, to be honest I looked everywhere, and I couldn't find it. But in interactive mode I do not have any leaks. Hopefully I do not lose points, I tried to look up what could cause valgrind to print the address of my functions and I couldn't find a specific reason. Also, I have some warnings when I compile but I hope I do not get marked off for that because my code still works correctly.

## **Conclusion**

In conclusion, I actually enjoyed this project. I just wish I didn't have to look for those two bugs I talked about in my Implantation, it took me 3 days and I had to ask professor Malony for help to find a bug. As I said before, the labs having part of the homework was very nice and helped me be confident throughout the project. This helped with debugging a lot. This project helped me understand how a shell works and I think this project should be what everyone has to do. One thing that I would change would be the error handling. That part I found fun at first but after implanting that everywhere I found it useless just repeating everything. Maybe I could have done the error handling in a more efficient way. Overall I enjoyed this project.