Name: Aleksandrs Lukjanovs

ID: M00674847

Course: 2020-21 CST2555 Operating Systems and Computer Networks

CompLab1/01 Tutor Name: Ahmed Eissa

Files: myshell.cpp , functions.cpp , makefile , readme.txt , documentation.docx

Video Demo: <https://youtu.be/vXja5DK6u1o>

**Coursework Readme File**

What have you done: I have done every task listed by the coursework.

What resources have been used: Large variety of random website mostly reddit for very specific problems and google to find out certain commands and ideas.

Which programming language: C++

**Explanation:**

To start my program consists of two large parts:

1. My functions.

2. My main which mainly houses my loop.

1.1.CB\_Request: Takes the user input then brakes it. This function firstly breaks down by characters looking for spaces then uses the space as a cut off point and it uses multiple vectors for either character accumulation or to save a set of characters.

1.2.TTT\_Request: It is similar to my CB\_Request by function but it's purpose is to break down the input and either add layers or remove them from directory links such as “/home/username/”.

2.1.Definitions: Some basic definitions for temp use or for long term storage of variables.

2.2.THE LOOP: This is where the magic happens. I do realize there were better ways to do this now but how this works is straightforward. It simply runs my output from CB\_Request through a large amount of if's which check if it's a command, and if it is in a correct format. Then it does appropriate action normally involving a call to the filesystem.

2.3.The pass through: Very straightforward as it simply lets anything that does not match my commands pass to the system.

**Difficulties Encountered:**

It turned out that the notes provided with the files were far more difficult than expected and I just did not have enough time to implement them properly. I struggled to understand them and C++ often gave unusual errors which were very time consuming to de-bugg.

**You can use the commands:**

* cd <directory> : to change directory. If the argument is not present, it will report the current directory. If the directory does not exist an appropriate error will be reported. This command will also change the PWD environment variable.
* cls : to clear the screen.
* dir <directory> : lists the contents of the directory.
* copy <source> <destination> : copies the <source> folder to <destination>.
* print <comment> : displays the commend on the display.
* md <directory> : creates a directory.
* rd <directory> : removes the folder if the folder is empty, will display error message if it is not.
* quit : exits the program and allows normal use of the command line.

**Resources Used:**

1. Stack
2. Google