Kurtis Chan
Github: kchanw
ID: 918139175
CSC415-01 Operating Systems

Assignment 3 – Simple Shell

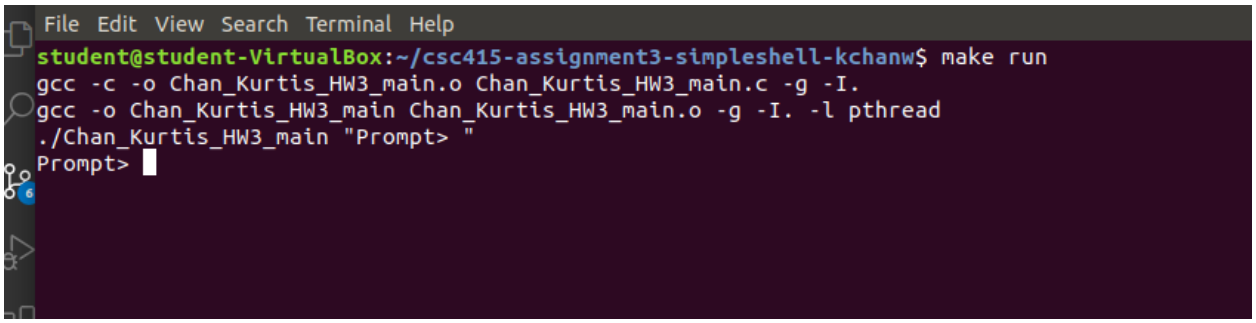
Description:
Writing a simple shell by using fork() and execvp() to handle commands. must take in a line of text and tokenize it to be ready for execvp(). Delim by space, tab, newline(whitespaces).Need to handle EOF and blank spaces.

Approach / What I Did:
First I went to look at how to use execvp() at this website that was posted on slack and discord which helped a lot on what needed to be passed and what it returns. Next to get a line of text, I needed to learn to use getline(), where the first argument is the pointer in my case buffer, then the size of the buffer, then stdin which is the file descriptor. This ties to the feof() where I saw at this website. Using feof will help determining when you reach the end of a file so it doesn't go forever. First to handle the input of a prompt, use argv[1] the second index to get what the prompt was. If there wasn't any prompt, then default to ">". Next will be making the loop for the shell which will be infinite till exit() is called. Next I used the slides for reference on the switch case to handle when the fork returns 0(the child), -1(error), positive number(parent), where the child will do execvp and output an error if there is one, error if fork fails and the parent where it shows the child pid, and exit status. I also put a strcmp for when "exit" is entered so it can exit the shell. Before that there is a wait till the child exits. Finally at the end, I set the argList and buffer to null, then free the allocated memory.

Issues and Resolutions:
The first problem was figuring out how to get input from a line, at first I thought it was in the main, but realized that would not work since argv would not be passed again, so I looked up the getline() function here. A problem I was stuck on was how to compare the command to the string "exit". It gave me an error where they could not be compared. Someone in the discord suggested using strcmp for it and once I looked up how to use it on geeksforgeeks, it helped compare the way I wanted it to.

Analysis: N/A

Screen shot of compilation:



Screen shot(s) of the execution of the program: