Names:
Shivani Murali
Lucas Grant

How to compile program:

- Enter 'make' into command line
- Execute using './mysh'

Files the program contains:

finalScript: please refer to this file to view all compilation of the test Runs that we used to test our code. This includes test cases for alias and the extra credit.

sh.c: contains functionality for all commands of the shell, calls fork(), execvp(), and wait()
We decided to use this code structure and leave the commands in this file because we documented each function such that the program is easily readable. Please refer to the documentation in the file to learn more about the functionality of each command.

Sh.h: Header file for sh

Makefile: used to compile the program

Sleep.c: use this file to test extra credit. Remember to compile this file before running ./mysh
Have to use gcc –o sleep sleep.c to compile it. When ./sleep is called after compiling ./mysh a child process will be created but it will be killed after 10 seconds because it takes too long to execute.

Get_path.c: Used to search path by looping through the path stored as a linked list

Get_path.h: Header file for get_path

Main.c: compiles the shell

Other Comments Regarding the Shell:

In our test cases instead of running /usr/bin/ls -l –g and /usr/bin/ls –l
We ran /bin/ls –l –g and /bin/ls –l because those were the directories that existed.

All built-in commands work according to their appropriate descriptions. Our code also checks for access if it is an absolute path. If it is neither a built in command or an absolute path, then our program finds the path and executes it. The * and ? wildcard charcters have been implemented as well. When Ctrl-D is entered our shell ignores the EOF char and prints a new prompt. When Ctrl-C or Ctrl-Z is pressed, our shell catched the signal and ignores it.

Please refer to the finalScript file to look at the test cases we ran for our program and sh.c to look at documentation.