CS 341
OS Assignment
Shell Implementation

Group 26
Ankit Kumar Singh - 150101086
Sudhanshu Ranjan - 150101076
Shashank Huddedar - 150101085

The Project consists of following files:

1. splitter.c
   splitter.h

   Files consists of function defination and declaration of str_split(char *) to split the given command into program name and arguments.

2. history.c
   history.h

   Files consists of functions defination and declaration of history(int n) that prints the last n commands and issue(int n) that runs the nth command.

3. helper.c
   helper.h

   Files consists of following functions:

   i. help() : to print all commands available
   ii. cd(char * s) : to change the present working directory to "s" if specified otherwise to $HOME env variable
   iii. ls(char *dir) : to print the files present in directory "dir" if specified else in present working directory.
   iv. signal_handler(): to handle the SIGALRM signal needed to kill the child after n seconds.

4. main.c

   Files takes in the input , process it and runs the corresponding function as per the command entered. Also it contains the header

5. Makefile File contains commands that helps in compling and linking all the specified source files together.

Commands Available

1. cd dir_name : Changes current directory if user has appropriate permissions
2. ls dir_name : Lists information about files in the current directory
3. rm : Deletes indicated files. Supports options –r, -f, -v
4. history n : Prints the most recent n commands issued by the numbers. If n is omitted, prints all commands issued by the user
5. issue n : Issues the nth command in the history once again.
6. <prog_name>: Creates a child process to run <prog_name>. Supports the redirection operators > and < to redirect the input and ouput of the program to indicated files.
7. exit : Exits the shell

8. rmexcept : Removes all files except those in list of files from the current directory.
9. <prog_name> m : Creates a child process to execute program_name, but aborts the process if it does not complete its operation in m seconds

How to Compile and Run?

1. To compile the code go to the project root directory. Run **make** command.

2. To run the code. Open a terminal. Run following command :

./shell_basic

this will start the terminal. Type "**help**" for list of all commands

Key Features

1. rmexcept – deleting all the files except one specified in cwd. This works recursively for the files even within the folders of cwd.

2. program_name m – runs the program for maximum of m seconds. This issues a SIGALRM after m seconds.

3. Both indirection and outdirection works simultaneously.

4. Terminal enviroment is made interactive using some colors(Traditional terminal like look).

Issues / Bugs

1. History file may get corrupted due to unhandled cases
2. Up arrow key does not gived previous history.
3. Tab completion does not work (Not implemented)

Contributions:

1. 150101085 : Worked together and did almost everything.
2. 150101076 : Worked together and did almost everything.
3. 150101086 : Worked together and did almost everything.