A simple Linux shell is written in C language. It's an Assignment required for the Operating System Course at IIIT-Delhi.
Professor: Sambuddho Chakravarty.

I named my Linux shell that is Wimix.
And I used the color code to put some highlighted text to give the user a good feeling about the Linux shell.

The Options for the shell Commands are:

1) hello
2) cd
3) pwd
4) echo
5) ls
6) cat
7) mkdir
8) date
9) rm
10) exit
11) help

* you can also press ↑ & ↓ while running “wimix” to execute any older command, which is done by add_history.

* you can start by typing hello.

With the help of “fork()/execl()” system call and “thread/system(),” the call is responsible for the process execution.
if a user wants to run any external command using thread, then he/she needs to start with the command preceding with &t.

>>Basically, the shell consists of two sub-commands to provide the functionality to the user.

>>the cd command moves from one directory to another.
the following two sub-commands for cd are:

1) cd ..
2) cd {PATH}

>>the echo command is used to print a string or ability of any other method4
by just typing echo {String}.

>>the pwd is used to print the live directory where the current user is in.

>>the ls command is used to peek at the files in a particular directory.
I code in such a way that the user can find some differences from the type of a file.
such as a directory inside a directory in red color.
the following two sub-commands for ls are:
1) ls -a: used to display hidden files.
2) ls -r: used to display files in reverse order.

>>the cat command is used to view the file content with the help of the shell.
1) cat -s {file} will suppress the repeatedly empty lines
2) cat -n {file} will show the content of the file preceding with line number

>>the mkdir is used to make directories.
1) mkdir -v: will pop up a message when a directory is ready.
2) mkdir -m used to provide some read/ write access to different groups.

>>the rm command is used to remove the file from the directory or a directory inside a directory.
1) rm -r: will delete the directory recursively.
2) rm -i: will delete but first ask you "whether you want to delete or not.

>>the date command is used to print the date. I selected two subcommands such as -I and -R; basically, they will print the date in different formats.

>>Assumptions: similar to the original UNIX shell.
if a false command is entered, then it shows some error.

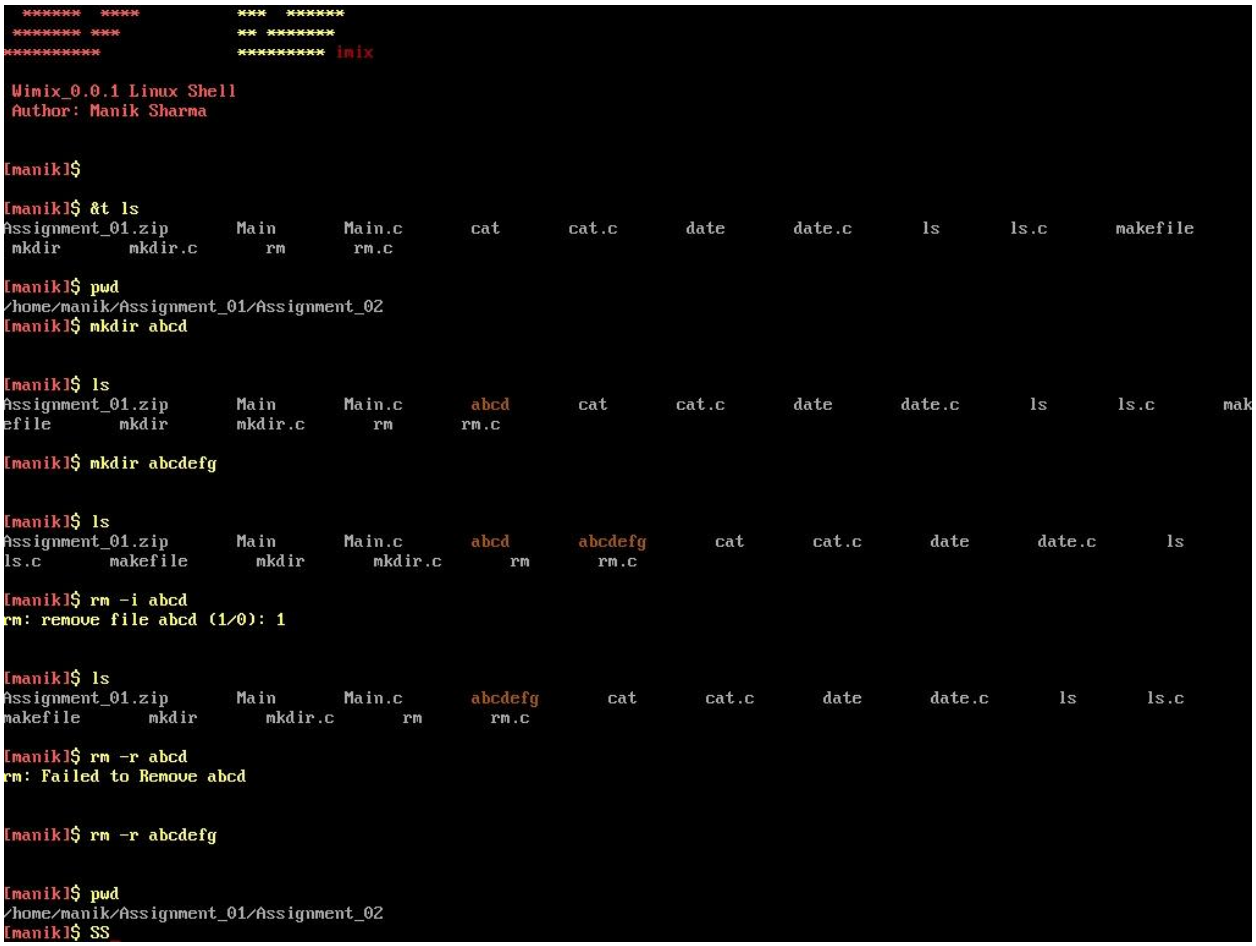
and if a user uses the cd command, then he needs to use
the pwd to check the directory.

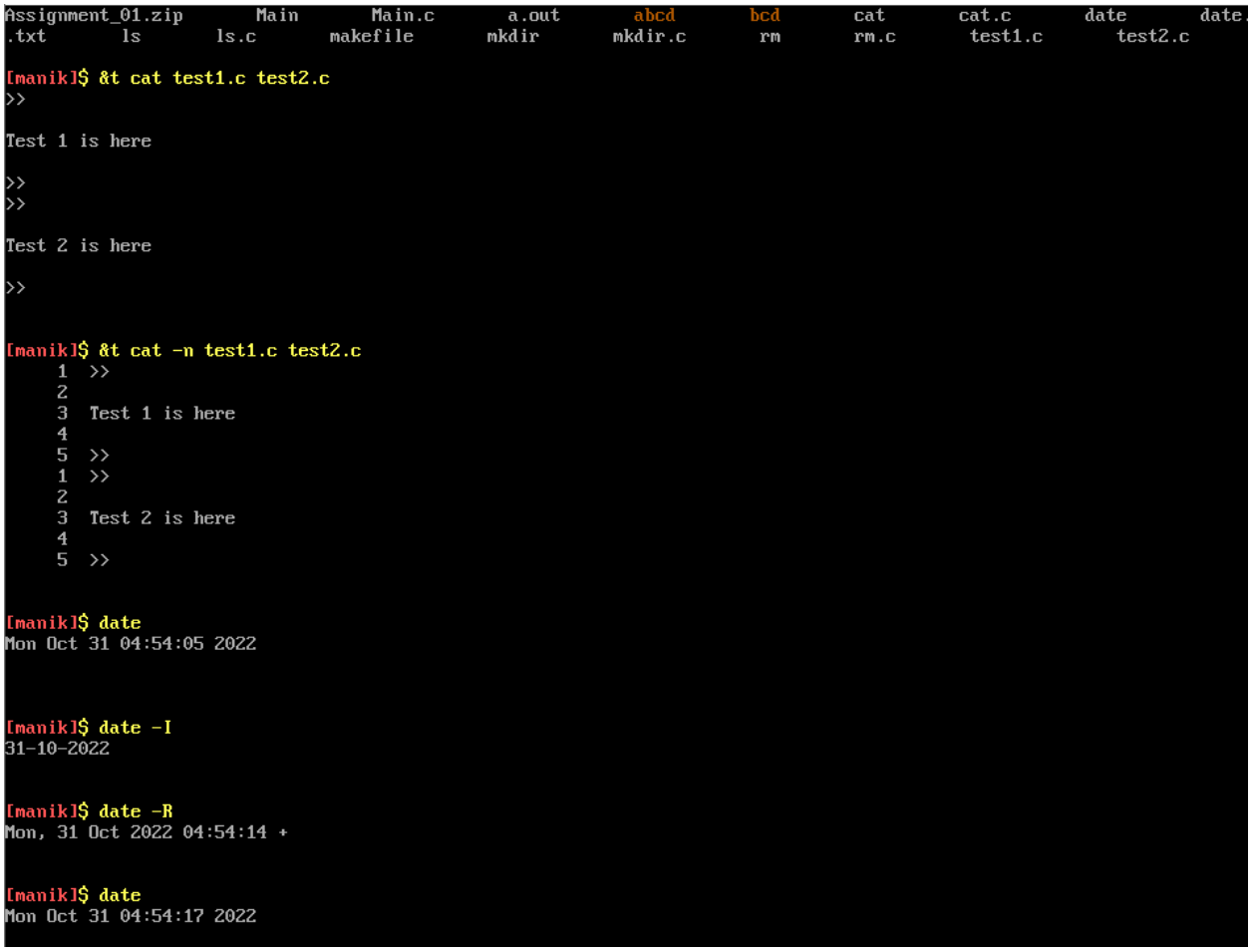
the system needs to pass a directory to check the file inside this directory.
and that is why the ls command is different from other external commands.

Users must use the make command in that directory where the original file is ready. Otherwise, the byte code will not generate.

When a user uses the ls -a and ls -r then there is a Main.c and byte file for the Main.c which will not count to find the list of files in ascending order or in descending order in a particular directory.

>> Test Cases


Wimix_0.0.1 Linux Shell
Author: Manik Sharma
[manik]$
[manik]$ &t ls
Assignment_01.zip Main Main.c cat cat.c date date.c ls ls.c makefile
mkdir mkdir.c rm rm.c
[manik]$ pwd
/home/manik/Assignment_01/Assignment_02
[manik]$ mkdir abcd
[manik]$ ls
Assignment_01.zip Main Main.c abcd cat cat.c date date.c ls ls.c mak
efile mkdir mkdir.c rm rm.c
[manik]$ mkdir abcdefg
[manik]$ ls
Assignment_01.zip Main Main.c abcd abcdefg cat cat.c date date.c ls
ls.c makefile mkdir mkdir.c rm rm.c
[manik]$ rm -i abcd
rm: remove file abcd (1/0): 1
[manik]$ ls
Assignment_01.zip Main Main.c abcdefg cat cat.c date date.c ls ls.c
makefile mkdir mkdir.c rm rm.c
[manik]$ rm -r abcd
rm: Failed to Remove abcd
[manik]$ rm -r abcdefg
[manik]$ pwd
/home/manik/Assignment_01/Assignment_02
[manik]$ SS



If you want to see the functionalities of the shell, along with some design requirements, look at the doc/ folder.

To compile the shell, run the following command:

make