

Write up Mannual

How to operate :

1. Use make command to run the program on the terminal.
2. In this shell of mine as instructed I'm using 4 internal commands and 5 external command.
3. Internal commands are actually defined inside main.c file and they are called by their respective main function.
4. To use external command in Question no. 1, first I used fork() command to create child process and then I used execv() command (which is actually belonged to the exec family) to run the executable file placed in its respective directory only.
5. To use external commands in Question no. 2, in case of fork() I have used pthread_create as well as pthread join and then I used system() command in order to call the executable file in place of execv() that we have used in the question 1

Description of the commands that I have used:

External Commands:-

1.cat :

1) cat -n filename1 filename2 filename3

Functionality, -> It reads data from the file and gives their content as output with the line number at start of each line in the file:

ex: cat -n geeks.txt

1 This is geeks

2 A unique array

2) cat filename1 filename2 filename3

Functionality, -> It reads data from the file and gives their content as output.

ex: cat geeks.txt hat.txt

This is geeks

A unique array

Errors: 1. Invalid input to execute the above command syntactically.
2. In case no such inputted filename exists in that directory.

2.date :

1) date (no option)

Functionality, -> With no options, the date command displays the current date and time, including the abbreviated day name, abbreviated month name, day of the month, the time separated by colons, the time zone name, and the year.

ex: date

Tue Oct 10 22:55:01 IST 2017

Note : Here unix system is configured in Indian Standard Time.

2) date -u

Functionality, -> With -u option this command displays the time in GMT(Greenwich Mean Time)/UTC(Coordinated Universal Time) time zone.

ex: date -u

Wed Oct 11 06:11:31 UTC 2017

Errors: Invalid input to execute the above command syntactically.

3.rm :

1) rm -i filename1 filename2 filename3

Functionality, -> the -i option makes the command ask the user for confirmation before removing each file, you have to press **y** for confirm deletion, any other key leaves the file un-deleted.

ex: rm -i d.txt

rm: remove regular empty file 'd.txt'? y

2) rm -f filename1 filename2 filename3

Functionality, -> **-f (Force Deletion):** rm prompts for confirmation removal if a file is **write protected**. The **-f** option overrides this minor protection and removes the file forcefully.
ex: rm -f e.txt

Errors: 1. Invalid input to execute the above command syntactically.
2. Incase no such inputted filename exists or due to some protected file or something in that directory.

4.mkdir :

1) mkdir -v directoryname1 directoryname2 directoryname3.....

Functionality, -> **-v or -verbose:** It displays a message for every directory created.
ex: mkdir -v a b c d

Directory: 'a' has been created successfully
Directory: 'b' has been created successfully
Directory: 'c' has been created successfully
Directory: 'd' has been created successfully

2) mkdir -p directoryname1 directoryname2 directoryname3.....

Functionality, -> **-p** A flag which enables the command to create parent directories as necessary. If the directories exist, no error is specified
ex: mkdir -p a b c d a

Errors: 1. Invalid input to execute the above command syntactically.
2. Incase inputted directory name already exist or some internal error.

5.ls :

1) ls (no option)

Functionality, -> **ls (no option)** is a Linux shell command that lists directory contents of files and directories space separated.
ex: ls

Makefile b	cat.c	date.c	hup	k	ls.c
mkdir	rm.c.	a	c	d	hupf
k.c	main	mkdir.c	a.out	cat	date
f	ls	main.c	rm		

2) ls -l

Functionality, -> **ls -l** is a Linux shell command that lists directory contents of files and directories that displays One File Per Line
ex:ls -l

Makefile b
cat.c
date.c
hup
k
ls.c
mkdir
rm.c.
a

3) ls -m

Functionality, -> **ls -m** is a Linux shell command that lists directory contents of files and directories comma separated.
ex: ls -m

Makefile, a, a.out, b, main, main.c, mkdir, mkdir.c, rm, rm.c

Errors: 1. Invalid input to execute the above command syntactically.

Important note regarding question 2:

Here in order to implement command using pthread user need to write "&t" in input after every external command initials. Eg: ls &t -l

Internal Commands:-

1.cd :

1) cd [directory_name or directory_path]

Functionality, -> **cd** command in linux known as change directory command. It is used to change current working directory.
ex: cd vidur

2) cd or cd ~

Functionality, -> this command is used to change directory to the home directory
ex: cd ~
ex: cd

Errors: 1. Invalid input to execute the above command syntactically.
2. Incase inputted directory name or path does not exist or typed wrong during input.

2.pwd :

1) pwd (no option)

Functionality, -> The pwd command writes to standard output the full path name of your current directory (from the root directory).also called built-in pwd
ex:pwd

/Users/local/Ass_1_os

2) pwd -L

Functionality, -> Prints the symbolic path of the current working directory
ex:pwd -L

/Users/local/Ass_1_os

3) pwd --help

Functionality, -> Prints all details about pwd command
ex:pwd --help
pwd: pwd [-L]
Print the name of the current working directory.
Options:
-L print the value of SPWD if it names the current working directory.

Errors: 1. Invalid input to execute the above command syntactically.

3.echo :

1) echo [string]

Assumption, -> Here we have assumed that no inverted commas are required in this during typing the string that we need to print on the shell.

Functionality, -> **echo** command in linux is used to display line of text/string that are passed as an argument and is used in shell to output status text to the screen or a file.
ex: echo Vidur is a student

Vidur is a student

2) echo *

Functionality, -> this command will print all files/folders, similar to ls command
ex: echo *

Makefile b cat.c date.c hup k ls.c

Errors: 1. Invalid input to execute the above command syntactically.

Assumption:

In my code I assume that the if input is entered which is space separated then it will assume that 2 different input is given & then corresponding command is executed.

Testcase:

```
ls
mkdir -p a
ls -l
mkdir -v as a b
ls -m
cd as
pwd
cd ..
cd ~
pwd -L
cd Codeforces
cd
cd holae
Ghcjcgjhj
pwd --help
cd Codeforces
ls
cat -n c.txt hpa.txt
cat c.txt
rm -f c.txt a.txt b.txt
rm -i d.txt ks.txt
date
date -u
cd 2021364_VIDUR
echo hello how are you vidur
echo *
exit
```

Testcase &t:

```
ls &t
mkdir &t -p a
ls &t -l
mkdir &t -v as a b
ls &t -m
cd as
pwd
cd ..
cd ~
pwd -L
cd Codeforces
cd
cd holae
Ghcjcgjhj
pwd --help
cd Codeforces
ls &t
cat &t -n c.txt hpa.txt
cat &t c.txt
rm &t -f c.txt a.txt b.txt
rm &t -i d.txt ks.txt
date &t
date &t -u
cd 2021364_VIDUR
echo hello how are you vidur
echo *
exit
```