

Os Assignment-1

The shell has 1 files one where every external command is run via fork()/execl() while other where every external command is called via pthread_create/System()

And switching term is &t at the end of the command

if&t is added the program runs via threading else via fork()

Threading achieved via pthread_create && pthread_cancel

And foring achieved via wait() and fork()

The shell implements 3 major internal commands being :-

1) cd (Change Directory)

1) it changes the directory the program is currently on to another directory asked by the programmer/tester

2) this implements two major types types being base, -p and -l

1) -l

It works like base cd and changes the directory as asked by user

2)-p

It first takes you to home directory and then absolutely follows the path of file

3) edge cases handled:-

1) If incase no file name was inputted

2) Or no file of that particular inputted directory existed or is private.

4) C functions used :- chdir() && getenv()

2) echo

1) Prints the input given to it

2) It implements base,-n and -e

a) -e

It ignores all the spaces between lines

b) -n

Ignores all trailing new lines

3) Edge cases handled are: -

1) to differentiate between th length of string and spaces between the string for each case mentioned

4) C functions used :- character and string printing

3) pwd

1) It implements -p and -l

a) -l

Prints working directory ignoring symlinks

b)-p

Prints actual current working directory

2) Edge cases handled

If not any random command is written after pwd command to break the program in any form

3) C functions used :- realpath() && getcwd()

External commands: -

LS

1) Implements ls -a and ls -r

ls -r

1) It prints all the directory in reverse order

ls -a

1) Prints all hidden directories

Edge cases:

For a wrong command

For a invalid command

For a invalid file

3) C functions used :- scandir() && struct dirent

Mkdir

1) implements -m and -v

1) -m allows to create files with different permission as inputted by the user where as -v and base mkdir allows to form the 0777 permission code directory

2) Edge Cases

1) no file is provided for after mkdir command

2) INVALID FILE TYPE IS PROVIDED ETC

3) C functions used :- mkdir()

Rm

1) Implements -v and -d

1) -d helps in removing empty directories

2) -v helps in removing a directory and providing an output for the same

2) Edge cases handled:-

Invalid file name is given

Invalid command followed by "rm"

3) C functions used :- unlink() && rmdir()

CAT

1) Implements -n and -E

1) -E prints dollar after every \n

2) -n prints every line along side the line number

2) Edge Cases

1) Handles file opening errors

- 2) File closing errors
- 3) Wrong file error
- 4) Invalid command error
- 3) C functions used :- write() && open()

Date

- 1) Implements BASE,-r and -u
 - 1) -u displays time in GMT
 - 2) displays date in day,month,time UTC ,yyyy format where as base provides month, date
- 2) Edgecases handled :-
 - 1) Invalid command after date prompt
 - 2) Invalid type of command
- 3) C functions used :- time() , localtime(), shrttime();