Unix Programming
Assignment - 1

1. What are the different types of files in UNIX?

The 3 categories of the files are :-

(i) Ordinary files. (or regular files)
(ii) Directory files (iii) Device files

(i) ordinary files :- It contains only data as a stream of characters.

→ Ordinary files is a file on the system that contains data, text or program instructions.

→ An ordinary data files can be either a text file or binary file.

→ Text file contains only printable characters & you can view & edite them

→ A binary file contains both printable & non printable characters that covers the entire ASCII range.

→ Hidden files : an invisible files is one the first character of which is the dot or the period character (.)

(ii) Directory files

→ A directory files contains no data but keeps details of the files & subdirectories that it contains.

→ A directory files contains one entry for every files & subdirectory that it houses.

- Each entry has 2 components namely.
  ⓐ File name & ⓑ unique identification number of the file or directory.
- Unix directories are equivalent to windows folders

* Device file:
* All the operations on the devices are performed by reading or writing the file representing the devices.
* Advantage of device file is that some of the commands used to access an ordinary file also work with device file.
* Device file name are generally found in a single directory structure / dev.
* It is the attributes of the file that entirely govern the operation of the device.

(7) Explain the shell interpretive lifecycle.

The shell performs following activities in its interpretive cycle:

(i) The shell :- issues the prompt ($) & waits for the enter a command (like ls chap *)

(ii) After a command is entered, the shell scans the command line for meta characters like ls chap* & expands the abbervations to recreate a simplified command line (ls chap1 chap2)

(iii) Then the shell passes the command line to the kernel for execution & waits for the command to complete its task.

(iv) After the command is executed, the shell issues the prommpt ($) again & waits for the user to enter a next command.

3. Explain about absolute & relative path name.

→ A pathname is a text string by made up of one or more names separated by a '/'.

* A pathname specifies how (.) to traverse (navigate) the hierarchial directory names in the file system to reach some destination object

Absolute pathname:

→ An absolute pathname begins with a slash (/).

→ The absolute path defines the location of on directory or a file from the root file system (/).

→ The absolute path contains the full path of to the directory or file.

Relative pathname.

→ The relative pathname do not begin with "/"

→ Specify the location relative to you current working directory.

→ . (a single dot) - this represent the current directory.

→ .. (2 dots) - this represents the parent directory.

Expln

date command can be executed in 2 ways.

using absolute pathname:

$ /bin/date

Thu Oct 15 10:20:29 IST 2020

using relative

$ date.

Thu Oct 15 10:20:29 IST 2020.

4. Explain file current permission.
rw-r-xr-- specify the chmod expression required to change them for the following.

(i) Rwxrwxrwx

Relative permissions.

chmod u+xg+wo+t wr demofile

Absolute permissions

chmod 777 demofile

(ii) r--x---

Relative permissions

chmod u-wg-x.o-r demofile.

Absolute permissions

chmod 440 demofile.

(iii) ----------

Relative permission:

chmod u-rw g-rx o-r demofile

Absolute permission.

chmod on o demofile

(v) ---r--R--.

Relative permission: chmod.

Absolute permission: chmod.

5). Briefly describe the following commands cat, cp, mv, rm wc, od.

→ * cat:- This command can also accept numbers than one

3.

display the content of a file on the terms

Syntax :- cat FILENAME

Case 1 :- This command can also accept numbu than one file as argument.

Example :- $ cat P1.C P2.C

→ Here the content of the second file is shown immediatly after the first file.

→ So, this command concatenate 2 files, hence its name (cat)

Case -2 :-

→ This command can also be used to create a new file

→ Syntax :-

cat > FILE NAME

Example :-

$ cat > P3.C welcome

TO SHELL // contents of P3.C

[ctrl-d] // ernicates P3.C

+ CP : This command is used to copy a file (s) from one location to acther location. It creates an exact image of the file on the disk with a different name

Syntax :- cp source-file destination-file.

Eg :- $ cp file1 file2 // copies content of file 1 to file 2

+ mv : This command renames or moves files

Case 1 : This command can be used to rename a file in the current directory

Syntax :-

mv OLD FILENAME NEWFILENAME

→ This command can also be used to move a group of files to a directory

* rm:-

This command can be used to delete a file.

Syntax:-

rm FILENAME

Example:-

$ rm FILE //deletes FILE

$ rm FILE1 FILE2 FILE3 //deletes 3 files.

* wc

→ This command can be used to get a count of the total number of lines words & characters contained in a file.

Syntax:-

wc FILE NAME

od:- This command can be used to display the content of executable file in a ASCII octal form

Ex :- $ cat P1.obj

abcd efgh //content of file P1.obj

abcd efgh