PRACTICAL – 1

2CEIT104:OPERATING SYSTEM

1. Execute following Linux commands and describe the output

TOUCH	CAT	LS	MKDIR
RMDIR	CD	CLEAR	CP
CAL	HISTORY	CHMOD	UMASK
HEAD	TAIL	DATE	EXPR
WHO	UNAME	FINGER	CMP
COMM	SORT	SPELL	WC
TYPE	TTY	ECHO	MAN
MORE	PASSWD	PWD	GREP
PS	RM	SET	CUT
READ	JOBS	AWK	LN
ENV	KILL	ALIAS	DIFF
LOCATE	FIND	INFO	

1.Touch :- The touch command is used to create empty files.We can create multiple empty files by executing it once.

Syntax - touch<filename>

2. Cat :- The cat command is a multi- purpose utility in the Linux system. It can be used to create a file , display content of the file , copy the content of one file to another file , and more.

Syntax - cat[OPTION]...[FILE]..

3. Ls :- The ls command is used to display a list of content of a directory

Synatax-ls

4. Mkdir :- It is use to make the new directory.

Syntax- mkdir<directoryname>

5. Rmdir :- It is use to remove the directory.

syntax-rmdir<directoyname>

6. Cd :- The cd command is used to change the current directory.

Syntax-cd <directoryname>

7. Clear :- Linux **clear** command is used to clear the term in a screen.

Syntax-clear

8. **CP** :- The cp command is used to copy a file or directory.

Syntax-cp<existing file name><new file name>

9. Cal :-The cal command is used to display the current month's calendar with the current date highlighted.

Syntax-cal

10. **History** :-It show all performed command until. ***Syntax-history***

11. **Chmod** :- In Unix-like operating systems, the **chmod** command is used to change the access mode of a file .The name is an abbreviation of **change mode**.

12. **Umask** :-The syntax of umask command is very simple where e just provide the permissions.

Synax-umask PERMISSIONS

13. Head :-The head command is used to display the content of a file. It displays the first

10 linesofafile.

Syntax:head<filename>

14. Tail :- The tailcommand is similar to the head command. The difference between both commands is that it displays the lastten lines of the file content. It is useful for reading the error message.

Syntax-tail<filename>

15. **Date** :- The date command is used to display date , time , time zone , and more. ***Syntax:date***

16. **Expr** :- The **expr** command in Unix evaluates a given expression and displays its corresponding output .It is used for : Basic operations like addition ,subtraction , multiplication ,division , and modulu son integers. Evaluatingregularexpressions,stringoperationslikesubstring,lengthofstringsetc.

Syntax-$expr expression

17. **Options** :-It is used to show the version information. ***Syntax: $expr—version***

18. **Who** :- The who **command** prints a list of all currently logged in users.

19. **Uname** :- The command '*uname*' displays the information about the system. ***Syntax-uname [OPTION]***

20. **Finger** :- **Finger** command is a user information lookup command which gives details of all the users logged in.This tool is generally used by system administrators. It provides details like login name, user name, idle time,login time, and in some cases their email address even. This tool is similar to the Pinky tool but the Pinkytoolis just the light weight version of this tool.

Syntax-$fingeraman

21. **Cmp** :- cmp command in Linux/UNIX is used to compare the two files byte by byte and helps you to find out whether the two files are identical or not. ***Syntax-cmp[OPTION]...FILE1[FILE2[SKIP1[SKIP2]]]***

22. **Comm** :- The 'comm' command is used to compare two files or streams. By default,

It displays three columns, first displays non-matching items of the first file, second indicates The non-match ingitem of the second file, and the third column displays the matching items of both files.

Syntax-comm<file1><file2>

23. Sort :- The sort command is used to sort files in alphabetical order.

24. Spell :- It is a very minimalistic spell-checking program, based on the original UNIX spell checker. It readsthe contents of file FILE, word for word, checking them against its dictionary. If a word does not correspondwithanyofspell'sdictionarywords, thewordisprinted. ***Syntax-spell[option]file***

25. Wc :- The wc command is used to count the lines,words,and characters in a file. ***Syntax:wc<filename>***

26. Type :- The **type** command is used to describe how its argument would be translated if used as commands . It is also used to find out whether it is built-in or external binary file. ***Syntax-type [Options] command names $typetype***

27. Tty :- Linux operating systemrepresents everything in a file system, the hardware devicesthat we attach arealso represented as a file. The terminal is also represented as a file. There a command exists called **tty** whichdisplays information related to **terminal**. The **tty** command of terminal basically prints the file name of the terminal connected to standard input. **Tty** is short of teletype, but popular lyknown as a terminal it allows you to interact with the system by passing on the data (you input)to the system,and displaying the output.

28. Echo :- This command in linux is used to display line of text/string that are passed as an argument. This is abuilt in command that is mostly use din shell scripts and batch files to output satus text to the screen or a file. ***Syntax-echo“string”***

29. **Man** :- The man command gives users access to manual pages for command line \ utilityes and tools.Following is the syntax of this command: ***Syntax-man[command/toolname]***

30. **More** :- The more command is quite similar to the cat command, a sit is used to display the file content in the same way that the cat command does. The only difference between both commands is that, in case of larger files, the more command displays screenful output at a time.
In more command , the following keys are used to scroll the page:
ENTER key **:** To scroll down page by line.
Space bar **:** To move to the next page.
B key **:** To move to the previous page.
Key **:** To search the string.
Syntax:more<filename>

31. Psswd :- The passwd command is used to create and change the password for a user.
Syntax-passwd<username>

32.Pwd :- This stands for **P**rint **W**orking **D**irectory. It prints the path of the working directory ,starting from the root.
Syntax-pwd - L : Prints the symbolic path.
Pwd - P : Prints the actual path.

33. Grep :- The grep is the most powerful and used fil terinal Linux system. The 'grep' stands for" **global regular expression print**. "It is use ful for search in gthe content from a file. Generally, it is used with the pipe.
Syntax : command | grep <search Word>

34. **Ps** :-Linux provides us a utility called **ps** for viewing information related with the processes on a system which stands as abbreviation for **"Process Status".** ps command is used to list the currently running processes and their PIDs along with some other information depends on different options. It reads the process information from the virtual files in**/proc** file-system./proc contains virtual files, this is the reason it's referred as a virtual file system.
Syntax-ps[options]

35. **Rm** :- The rm command is used to remove a file.
Syntax-rm<filename>

36. **Set** :- Linux set command is used to set and unset certain flags or settings within the shell environment. These flags and settings determine the behavior of a defined script and help in executing the tasks with out facing any issue. The values of shell attributes and parameters can be changed or displayed by using the set command. ***Syntax-set[options]***

37. **Cut** :- The cut command is used to select a specific column of a file. The'-d' option is used as a delimiter, and it can be a space(''), a slash(/),a hyphen(-),or any thing else.
And, the'-f' option is used to specify a column number.
Syntax: cut-d(delimiter)-f (column Number)<file Name>

38. **Read** :- Linux read command is used to read the contents of a line into a variable. This is a built-in command for Linux systems. Therefore, we do not need to install any additional tools. It is an easy tool to take user input when creating abash script. It is a powerful utility and as important as echo command and positional parameter.
Syntax:read[options][name...]

39. **Job** :- The *jobs* command displays the status of jobs started in the current terminal window. Jobs are number ed starting from**1** for each session. The job ID numbers are used by some programs instead of **PIDs** (for example, by*f* and *bg* commands). ***Syntax-jobs***

40. **Awk** :- You can instruct AWK to print only certain columns from the input field.

41. **Ln** :- The *ln* command is used to create links between files. Before going into the application of the ln command in detail, please refer the below link for a clear understanding of the hard link and soft link in Linux. Hardand Soft Links in Linux ***Syntax : ln _test_dir/filename***

42. **Env** :- It is used to either print environment variables. It is also used to run a utility or command in a custom environment. In practice, env has another common use. It is often used by shell scripts to launch the correct interpreter. In this usage , the environment is typically not changed.

Syntax-env[OPTION]...[-][NAME=VALUE]...[COMMAND[ARG]...]

43. **Kill** :- The Kill command in unix or linux operating system is used to send a signal to the specified process or group. If we dont specify any signal, then the kill command passes the SIGTERM signal. We mostly use the kill command for terminating or killing a process. However we can also use the kill command for running as top ped process. ***Syntaxkill[-ssignal]pidkill-l***

44. **Alias** :- This command in structs the shell to replace one string with another String while executing the commands. ***Syntax-alias[-p][name[=value]...]***

45. **Dif** :- It stands for **difference**. This command is used to display the differences in the files by comparing the files line by line. Unlike its fellow members, cmp and comm, it tells us which lines in one file have is to be changed to make the two files identical. ***Syntax-diff[options]File1File2***

46. **Locate** :- *This* command in Linux is used to find the files by name. There is two most widely used file search in gutilities accessible to users are called find and *locate*. The *locate* utility works better and faster than *find* command counter part because in stead of searching the file system when a file search is initiated, it would look through a database. This database contains bits and parts of files and their corresponding paths on your system. By default,

locate command does not check whether the files found in the database still exist and never reports files created after the most recent update of the relevant database. ***Syntax-locate[OPTION]...PATTERN...***

47. **Find** :- The **find** command in UNIX is a command line utility for walking a file hierarchy. It can be used to find files and directories and perform subsequent operations on them. It supports searching by file, folder, name , creation date, modification date, owner and permissions. By using the '-exec' other UNIX commands can be executed on files or folders found.

Syntax-$find[wheretostartsearchingfrom] [expression determines what to find] [- options][what to find]

48. Info :- **info** command reads documentation in the *info* format. It will give detailed information for a command when compared with the man page. The pages are made using the *text info* tools because of which it can link with other pages , create menus and easy navigation.

Syntax-info[OPTION]...[MENU-ITEM...]