

OPERATING SYSTEM LAB MANUAL

Ex.No:1.a	BASICS OF UNIX COMMANDS
	INTRODUCTION TO UNIX

AIM:

To study about the basics of UNIX

UNIX:

It is a multi-user operating system. Developed at AT & T Bell Industries, USA in 1969.

Ken Thomson along with Dennis Ritchie developed it from MULTICS (Multiplexed Information and Computing Service) OS.

By 1980, UNIX had been completely rewritten using C language.

LINUX:

It is similar to UNIX, which is created by Linus Torvalds. All UNIX commands work in Linux. Linux is an open source software. The main feature of Linux is coexisting with other OS such as Windows and UNIX.

STRUCTURE OF A LINUX SYSTEM:

It consists of three parts.

- a) UNIX kernel
- b) Shells
- c) Tools and Applications

UNIX KERNEL:

Kernel is the core of the UNIX OS. It controls all tasks, schedules all processes and carries out all the functions of OS.

Decides when one program stops and another starts.

SHELL:

Shell is the command interpreter in the UNIX OS. It accepts commands from the user and analyses and interprets them.

Ex.No:1.b	BASICS OF UNIX COMMANDS
	BASIC UNIX COMMANDS

AIM:

To study of Basic UNIX Commands and various UNIX editors such as vi, ed, ex and EMACS.

CONTENT:

Note: Syn->Syntax

a) date

–used to check the date and time

Syn:\$date

Format	Purpose	Example	Result
+%m	To display only month	\$date+%m	06
+%h	To display month name	\$date+%h	June
+%d	To display day of month	\$date+%d	01
+%y	To display last two digits of years	\$date+%y	09
+%H	To display hours	\$date+%H	10
+%M	To display minutes	\$date+%M	45
+%S	To display seconds	\$date+%S	55

b) cal

–used to display the calendar

Syn:\$cal 2 2009

c)echo

–used to print the message on the screen.

Syn:\$echo “text”

d)ls

–used to list the files. Your files are kept in a directory.

Syn:\$ls-ls-s

All files (include files with prefix)

ls-l Lodash (provide file statistics)

ls-t Order by creation time

ls-u Sort by access time (or show when last accessed together with -l)

ls-s Order by size

ls-r Reverse order

ls-f Mark directories with /, executable with *, symbolic links with @, local sockets with =, named pipes (FIFOs) with

ls-s Show file size

ls-h “Human Readable”, show file size in Kilo Bytes & Mega Bytes (h can be used together with -l or)

ls[a-m]*List all the files whose name begin with alphabets From 'a' to 'm'
 ls[a]*List all the files whose name begins with 'a' or 'A'
 Eg:\$ls>my list Output of 'ls' command is stored to disk file named 'my list'

e)lp

–used to take printouts

Syn:\$lp filename

f)man

–used to provide manual help on every UNIX commands.

Syn:\$man unix command

\$man cat

g)who & whoami

–it displays data about all users who have logged into the system currently. The next command displays about current user only.

Syn:\$who\$whoami

h)uptime

–tells you how long the computer has been running since its last reboot or power-off.

Syn:\$uptime

i)uname

–it displays the system information such as hardware platform, system name and processor, OS type

Syn:\$uname–a

j)hostname

–displays and set system host name

Syn:\$ hostname

k)bc

–stands for 'best calculator'

```
$bc
10/2*3
15
```

```
$ bc
scale =1
2.25+1
3.35
quit
```

```
$ bc
ibase=2
obase=16
11010011
89275
1010
Ā
Quit
```

```
$ bc
sqrt(196)
14 quit
```

```
$bc
for(i=1;i<3;i=i+1)I
1
2
3 quit
```

```
$ bc-l
scale=2
s(3.14)
0
```

FILE MANIPULATION COMMANDS

a) **cat**—this create, view and concatenate files.

Creation:

Syn:\$cat>filename

Viewing:

Syn:\$cat filename

Add text to an existing file:

Syn:\$cat>>filename

Concatenate:

Syn:\$catfile1 file2>file3

\$catfile1 file2>>file3 (no over writing of file3)

b) **grep**—used to search a particular word or pattern related to that word from the file.

Syn:\$grep search word filename

Eg:\$grep anu student

c) **rm**—deletes a file from the file system

Syn:\$rm filename

d) **touch**—used to create a blank file.

Syn:\$touch file names

e) **cp**—copies the files or directories

Syn:\$cpsource file destination file

Eg:\$cp student stud

f) **mv**—to rename the file or directory

syn:\$mv old file new file

Eg:\$mv-i student student list(-i prompt when overwrite)

g) **cut**—it cuts or pickup a given number of character or fields of the file.

Syn:\$cut<option><filename>

Eg: \$cut -c filename

\$cut-c1-10emp

\$cut-f 3,6emp

\$ cut -f 3-6 emp

-c cutting columns

-f cutting fields

h) **head**—displays10 lines from the head(top)of a given file

Syn:\$head filename

Eg:\$head student

To display the top two lines:

Syn:\$head-2student

i)**tail**–displays last 10 lines of the file

Syn:\$tail filename

Eg:\$tail student

To display the bottom two lines;

Syn:\$ tail -2 student

j)**chmod**–used to change the permissions of a file or directory.

Syn:\$ch mod category operation permission file

Where, Category–is the user type

Operation–is used to assign or remove permission

Permission–is the type of permission

File–are used to assign or remove permission all

Examples:

\$chmodu-wx student

Removes write and execute permission for users

\$ch modu+rw,g+rwxstudent

Assigns read and write permission for users and groups

\$chmodg=rwx student

Assigns absolute permission for groups of all read, write and execute permissions

k)**wc**–it counts the number of lines, words, character in a specified file(s)
with the options as -l,-w,-c

Category	Operation	Permission
u– users	+assign	r– read
g–group	-remove	w– write
o– others	=assign absolutely	x–execute

Syn: \$wc -l filename

\$wc -w filename

\$wc -c filename