Name : Varad Girish Mashalkar

Roll Number : 33337

Batch : M11

Course : Operating Systems Laboratory

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Assignment** : 1A

**Problem Statement**

Study of Basic Linux Commands: echo, ls, read, cat, touch, test, loops, arithmetic comparison,

conditional loops, grep, sed etc.

**Theory and Execution**

* Linux Commands are case sensitive in nature.
* Format : <command\_name> <options> <file\_name1> <file\_name2>..
* The commands enable to interact with file structures, take conditional descisions, etc.
* Following Commands were studied and executed on Linux terminal.

**1) echo**

**Manual :**

ECHO(1) User Commands ECHO(1)

NAME

echo - display a line of text

SYNOPSIS

echo [SHORT-OPTION]... [STRING]...

echo LONG-OPTION

DESCRIPTION

Echo the STRING(s) to standard output.

-n do not output the trailing newline

-e enable interpretation of backslash escapes

-E disable interpretation of backslash escapes (default)

--help display this help and exit

--version

output version information and exit

If -e is in effect, the following sequences are recognized:

\\ backslash

\a alert (BEL)

\b backspace

\c produce no further output

\e escape

\f form feed

\n new line

\r carriage return

\t horizontal tab

\v vertical tab

\0NNN byte with octal value NNN (1 to 3 digits)

\xHH byte with hexadecimal value HH (1 to 2 digits)

NOTE: your shell may have its own version of echo, which usually super‐

sedes the version described here. Please refer to your shell's docu‐

mentation for details about the options it supports.

**Output :**

**varadmash@varadmash-G3-3590:~$ echo -e "Hello\nWorld"**

**Hello**

**World**

**2) ls**

**Manual**

LS(1) User Commands

NAME

ls - list directory contents

SYNOPSIS

ls [OPTION]... [FILE]...

DESCRIPTION

List information about the FILEs (the current directory by default). Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.

Mandatory arguments to long options are mandatory for short options too.

-a, --all

do not ignore entries starting with .

-A, --almost-all

do not list implied . and ..

--author

with -l, print the author of each file

-b, --escape

print C-style escapes for nongraphic characters

--block-size=SIZE

with -l, scale sizes by SIZE when printing them; e.g., '--block-size=M'; see SIZE format below

-B, --ignore-backups

do not list implied entries ending with ~

-c with -lt: sort by, and show, ctime (time of last modification of file status information); with -l: show ctime and sort by name; otherwise: sort by ctime, newest first

-C list entries by columns

--color[=WHEN]

colorize the output; WHEN can be 'always' (default if omitted), 'auto', or 'never'; more info below

-d, --directory

list directories themselves, not their contents

-D, --dired

generate output designed for Emacs' dired mode

-f do not sort, enable -aU, disable -ls --color

-F, --classify

append indicator (one of \*/=>@|) to entries

--file-type

likewise, except do not append '\*'

--format=WORD

across -x, commas -m, horizontal -x, long -l, single-column -1, verbose -l, vertical -C

--full-time

like -l --time-style=full-iso

-g like -l, but do not list owner

--group-directories-first

group directories before files;

can be augmented with a --sort option, but any use of --sort=none (-U) disables grouping

-G, --no-group

in a long listing, don't print group names

-h, --human-readable

with -l and -s, print sizes like 1K 234M 2G etc.

--si likewise, but use powers of 1000 not 1024

-H, --dereference-command-line

follow symbolic links listed on the command line

--dereference-command-line-symlink-to-dir

follow each command line symbolic link

that points to a directory

--hide=PATTERN

do not list implied entries matching shell PATTERN (overridden by -a or -A)

--hyperlink[=WHEN]

hyperlink file names; WHEN can be 'always' (default if omitted), 'auto', or 'never'

--indicator-style=WORD

append indicator with style WORD to entry names: none (default), slash (-p), file-type (--file-type), classify (-F)

-i, --inode

print the index number of each file

-I, --ignore=PATTERN

do not list implied entries matching shell PATTERN

-k, --kibibytes

default to 1024-byte blocks for disk usage; used only with -s and per directory totals

-l use a long listing format

-L, --dereference

when showing file information for a symbolic link, show information for the file the link references rather than for the link itself

-m fill width with a comma separated list of entries

-n, --numeric-uid-gid

like -l, but list numeric user and group Ids

-N, --literal

print entry names without quoting

-o like -l, but do not list group information

-p, --indicator-style=slash

append / indicator to directories

-q, --hide-control-chars

print ? instead of nongraphic characters

--show-control-chars

show nongraphic characters as-is (the default, unless program is 'ls' and output is a terminal)

-Q, --quote-name

enclose entry names in double quotes

--quoting-style=WORD

use quoting style WORD for entry names: literal, locale, shell, shell-always, shell-escape, shell-escape-always, c, escape (overrides QUOTING\_STYLE environment variable)

-r, --reverse

reverse order while sorting

-R, --recursive

list subdirectories recursively

-s, --size

print the allocated size of each file, in blocks

-S sort by file size, largest first

--sort=WORD

sort by WORD instead of name: none (-U), size (-S), time (-t), version (-v), extension (-X)

--time=WORD

with -l, show time as WORD instead of default modification time: atime or access or use (-u); ctime or status (-c); also use specified time as sort key if --sort=time (newest first)

--time-style=TIME\_STYLE

time/date format with -l; see TIME\_STYLE below

-t sort by modification time, newest first

-T, --tabsize=COLS

assume tab stops at each COLS instead of 8

-u with -lt: sort by, and show, access time; with -l: show access time and sort by name; otherwise: sort by access time, newest first

-U do not sort; list entries in directory order

-v natural sort of (version) numbers within text

-w, --width=COLS

set output width to COLS. 0 means no limit

-x list entries by lines instead of by columns

-X sort alphabetically by entry extension

-Z, --context

print any security context of each file

-1 list one file per line. Avoid '\n' with -q or -b

--help display this help and exit

--version

output version information and exit

The SIZE argument is an integer and optional unit (example: 10K is 10\*1024). Units are K,M,G,T,P,E,Z,Y (powers of 1024) or KB,MB,... (powers of 1000).

The TIME\_STYLE argument can be full-iso, long-iso, iso, locale, or +FORMAT. FORMAT is interpreted like in date(1). If FORMAT is FORMAT1<newline>FORMAT2, then FORMAT1 applies to non-recent

files and FORMAT2 to recent files. TIME\_STYLE prefixed with 'posix-' takes effect only outside the POSIX locale. Also the TIME\_STYLE environment variable sets the default style to use.

Using color to distinguish file types is disabled both by default and with --color=never. With --color=auto, ls emits color codes only when standard output is connected to a terminal. The

LS\_COLORS environment variable can change the settings. Use the dircolors command to set it.

**Output :**

varadmash@varadmash-G3-3590:~$ ls -l

total 78404

drwxrwxr-x 29 varadmash varadmash 4096 Aug 30 16:09 anaconda3

drwxrwxr-x 2 varadmash varadmash 4096 Aug 30 15:43 conda\_environment\_ymls

drwxrwxr-x 4 varadmash varadmash 4096 Aug 22 19:31 CPP

drwxr-xr-x 2 varadmash varadmash 4096 Sep 2 08:32 Desktop

drwxr-xr-x 2 varadmash varadmash 4096 Aug 16 06:34 Documents

drwxr-xr-x 2 varadmash varadmash 4096 Sep 4 10:11 Downloads

-rw-rw-r-- 1 varadmash varadmash 80213604 Jul 31 02:37 google-chrome-stable\_current\_amd64.deb

drwxrwxr-x 3 varadmash varadmash 4096 Aug 31 20:19 LP1\_lab

drwxrwxr-x 3 varadmash varadmash 4096 Aug 30 19:50 ML\_Notebooks

drwxr-xr-x 2 varadmash varadmash 4096 Aug 16 06:34 Music

drwxrwxr-x 6 varadmash varadmash 4096 Sep 1 14:12 OS\_Lab

drwxr-xr-x 3 varadmash varadmash 4096 Sep 3 14:26 Pictures

drwxr-xr-x 2 varadmash varadmash 4096 Aug 16 06:34 Public

drwxr-xr-x 3 varadmash varadmash 4096 Aug 15 21:38 snap

drwxr-xr-x 2 varadmash varadmash 4096 Aug 16 06:34 Templates

-rw-rw-r-- 1 varadmash varadmash 41 Aug 15 23:12 token.txt

drwxr-xr-x 2 varadmash varadmash 4096 Aug 16 06:34 Videos

drwxrwxr-x 3 varadmash varadmash 4096 Aug 15 22:41 web\_dev

**3) read**

**Manual :**

NAME

read - read from a file descriptor

SYNOPSIS

#include <unistd.h>

ssize\_t read(int fd, void \*buf, size\_t count);

DESCRIPTION

read() attempts to read up to count bytes from file descriptor fd into the buffer starting at buf.

On files that support seeking, the read operation commences at the file offset, and the file offset is incremented by the number of bytes read. If the file offset is at or past the end of

file, no bytes are read, and read() returns zero.

If count is zero, read() may detect the errors described below. In the absence of any errors, or if read() does not check for errors, a read() with a count of 0 returns zero and has no

other effects.

According to POSIX.1, if count is greater than SSIZE\_MAX, the result is implementation-defined; see NOTES for the upper limit on Linux.

RETURN VALUE

On success, the number of bytes read is returned (zero indicates end of file), and the file position is advanced by this number. It is not an error if this number is smaller than the number

of bytes requested; this may happen for example because fewer bytes are actually available right now (maybe because we were close to end-of-file, or because we are reading from a pipe, or

from a terminal), or because read() was interrupted by a signal. See also NOTES.

On error, -1 is returned, and errno is set appropriately. In this case, it is left unspecified whether the file position (if any) changes.

ERRORS

EAGAIN The file descriptor fd refers to a file other than a socket and has been marked nonblocking (O\_NONBLOCK), and the read would block. See open(2) for further details on the O\_NONBLOCK

flag.

EAGAIN or EWOULDBLOCK

The file descriptor fd refers to a socket and has been marked nonblocking (O\_NONBLOCK), and the read would block. POSIX.1-2001 allows either error to be returned for this case, and

does not require these constants to have the same value, so a portable application should check for both possibilities.

EBADF fd is not a valid file descriptor or is not open for reading.

EFAULT buf is outside your accessible address space.

EINTR The call was interrupted by a signal before any data was read; see signal(7).

EINVAL fd is attached to an object which is unsuitable for reading; or the file was opened with the O\_DIRECT flag, and either the address specified in buf, the value specified in count, or

the file offset is not suitably aligned.

EINVAL fd was created via a call to timerfd\_create(2) and the wrong size buffer was given to read(); see timerfd\_create(2) for further information.

EIO I/O error. This will happen for example when the process is in a background process group, tries to read from its controlling terminal, and either it is ignoring or blocking SIGTTIN

or its process group is orphaned. It may also occur when there is a low-level I/O error while reading from a disk or tape. A further possible cause of EIO on networked filesystems

is when an advisory lock had been taken out on the file descriptor and this lock has been lost. See the Lost locks section of fcntl(2) for further details.

EISDIR fd refers to a directory.

Other errors may occur, depending on the object connected to fd.

**Output :**

varadmash@varadmash-G3-3590:~$ read my\_name

Varad

varadmash@varadmash-G3-3590:~$ echo $my\_name

Varad

**4) cat**

**Manual :**

NAME

cat - concatenate files and print on the standard output

SYNOPSIS

cat [OPTION]... [FILE]...

DESCRIPTION

Concatenate FILE(s) to standard output.

With no FILE, or when FILE is -, read standard input.

-A, --show-all

equivalent to -vET

-b, --number-nonblank

number nonempty output lines, overrides -n

-e equivalent to -vE

-E, --show-ends

display $ at end of each line

-n, --number

number all output lines

-s, --squeeze-blank

suppress repeated empty output lines

-t equivalent to -vT

-T, --show-tabs

display TAB characters as ^I

-u (ignored)

-v, --show-nonprinting

use ^ and M- notation, except for LFD and TAB

--help display this help and exit

--version

output version information and exit

EXAMPLES

cat f - g

Output f's contents, then standard input, then g's contents.

cat Copy standard input to standard output.

**Output :**

varadmash@varadmash-G3-3590:~$ cat >> new\_file.txt

The

Quick

Brown

Fox

Jumped

Over

The

Fence

^Z

[1]+ Stopped cat >> new\_file.txt

varadmash@varadmash-G3-3590:~$ cat new\_file.txt

The

Quick

Brown

Fox

Jumped

Over

The

Fence

**5) touch**

**Manual**

TOUCH(1) User Commands TOUCH(1)

NAME

touch - change file timestamps

SYNOPSIS

touch [OPTION]... FILE...

DESCRIPTION

Update the access and modification times of each FILE to the current time.

A FILE argument that does not exist is created empty, unless -c or -h is supplied.

A FILE argument string of - is handled specially and causes touch to change the times of the file associated with standard output.

Mandatory arguments to long options are mandatory for short options too.

-a change only the access time

-c, --no-create

do not create any files

-d, --date=STRING

parse STRING and use it instead of current time

-f (ignored)

-h, --no-dereference

affect each symbolic link instead of any referenced file (useful only on systems that can change the timestamps of a symlink)

-m change only the modification time

-r, --reference=FILE

use this file's times instead of current time

-t STAMP

use [[CC]YY]MMDDhhmm[.ss] instead of current time

--time=WORD

change the specified time: WORD is access, atime, or use: equivalent to -a WORD is modify or mtime: equivalent to -m

--help display this help and exit

--version

output version information and exit

Note that the -d and -t options accept different time-date formats.

**Output :**

varadmash@varadmash-G3-3590:~$ touch hello\_world.txt

varadmash@varadmash-G3-3590:~$ ls

anaconda3 CPP Documents google-chrome-stable\_current\_amd64.deb LP1\_lab Music OS\_Lab Public Templates Videos

conda\_environment\_ymls Desktop Downloads hello\_world.txt ML\_Notebooks new\_file.txt Pictures snap token.txt web\_dev

**6) test**

**Output :**

AME

test - check file types and compare values

SYNOPSIS

test EXPRESSION

test

[ EXPRESSION ]

[ ]

[ OPTION

DESCRIPTION

Exit with the status determined by EXPRESSION.

--help display this help and exit

--version

output version information and exit

An omitted EXPRESSION defaults to false. Otherwise, EXPRESSION is true or false and sets exit status. It is one of:

( EXPRESSION )

EXPRESSION is true

! EXPRESSION

EXPRESSION is false

EXPRESSION1 -a EXPRESSION2

both EXPRESSION1 and EXPRESSION2 are true

EXPRESSION1 -o EXPRESSION2

either EXPRESSION1 or EXPRESSION2 is true

-n STRING

the length of STRING is nonzero

STRING equivalent to -n STRING

-z STRING

the length of STRING is zero

STRING1 = STRING2

the strings are equal

STRING1 != STRING2

the strings are not equal

INTEGER1 -eq INTEGER2

INTEGER1 is equal to INTEGER2

INTEGER1 -ge INTEGER2

INTEGER1 is greater than or equal to INTEGER2

INTEGER1 -gt INTEGER2

INTEGER1 is greater than INTEGER2

INTEGER1 -le INTEGER2

INTEGER1 is less than or equal to INTEGER2

INTEGER1 -lt INTEGER2

INTEGER1 is less than INTEGER2

INTEGER1 -ne INTEGER2

INTEGER1 is not equal to INTEGER2

FILE1 -ef FILE2

FILE1 and FILE2 have the same device and inode numbers

FILE1 -nt FILE2

FILE1 is newer (modification date) than FILE2

FILE1 -ot FILE2

FILE1 is older than FILE2

-b FILE

FILE exists and is block special

-c FILE

FILE exists and is character special

-d FILE

FILE exists and is a directory

-e FILE

FILE exists

-f FILE

FILE exists and is a regular file

-g FILE

FILE exists and is set-group-ID

-G FILE

FILE exists and is owned by the effective group ID

-h FILE

FILE exists and is a symbolic link (same as -L)

-k FILE

FILE exists and has its sticky bit set

-L FILE

FILE exists and is a symbolic link (same as -h)

-O FILE

FILE exists and is owned by the effective user ID

-p FILE

FILE exists and is a named pipe

-r FILE

FILE exists and read permission is granted

-s FILE

FILE exists and has a size greater than zero

-S FILE

FILE exists and is a socket

-t FD file descriptor FD is opened on a terminal

-u FILE

FILE exists and its set-user-ID bit is set

-w FILE

FILE exists and write permission is granted

-x FILE

FILE exists and execute (or search) permission is granted

Except for -h and -L, all FILE-related tests dereference symbolic links. Beware that parentheses need to be escaped (e.g., by backslashes) for shells. INTEGER may also be -l STRING, which

evaluates to the length of STRING.

NOTE: Binary -a and -o are inherently ambiguous. Use 'test EXPR1 && test EXPR2' or 'test EXPR1 || test EXPR2' instead.

NOTE: [ honors the --help and --version options, but test does not. test treats each of those as it treats any other nonempty STRING.

NOTE: your shell may have its own version of test and/or [, which usually supersedes the version described here. Please refer to your shell's documentation for details about the options it

supports.

**Output :**

varadmash@varadmash-G3-3590:~$ if [ 4 -gt 2 ]

> then

> echo "4 is greater than 2"

> else

> echo "2 is greater than equal to 4"

> fi

4 is greater than 2

**7) grep**

**Manual :**

NAME

grep, egrep, fgrep, rgrep - print lines that match patterns

SYNOPSIS

grep [OPTION...] PATTERNS [FILE...]

grep [OPTION...] -e PATTERNS ... [FILE...]

grep [OPTION...] -f PATTERN\_FILE ... [FILE...]

DESCRIPTION

grep searches for PATTERNS in each FILE. PATTERNS is one or more patterns separated by newline characters, and grep prints each line that matches a pattern. Typically PATTERNS should be

quoted when grep is used in a shell command.

A FILE of “-” stands for standard input. If no FILE is given, recursive searches examine the working directory, and nonrecursive searches read standard input.

In addition, the variant programs egrep, fgrep and rgrep are the same as grep -E, grep -F, and grep -r, respectively. These variants are deprecated, but are provided for backward

compatibility.

OPTIONS

Generic Program Information

--help Output a usage message and exit.

-V, --version

Output the version number of grep and exit.

Pattern Syntax

-E, --extended-regexp

Interpret PATTERNS as extended regular expressions (EREs, see below).

-F, --fixed-strings

Interpret PATTERNS as fixed strings, not regular expressions.

-G, --basic-regexp

Interpret PATTERNS as basic regular expressions (BREs, see below). This is the default.

-P, --perl-regexp

Interpret PATTERNS as Perl-compatible regular expressions (PCREs). This option is experimental when combined with the -z (--null-data) option, and grep -P may warn of unimplemented

features.

Matching Control

-e PATTERNS, --regexp=PATTERNS

Use PATTERNS as the patterns. If this option is used multiple times or is combined with the -f (--file) option, search for all patterns given. This option can be used to protect a

pattern beginning with “-”.

-f FILE, --file=FILE

Obtain patterns from FILE, one per line. If this option is used multiple times or is combined with the -e (--regexp) option, search for all patterns given. The empty file contains

zero patterns, and therefore matches nothing.

-i, --ignore-case

Ignore case distinctions in patterns and input data, so that characters that differ only in case match each other.

--no-ignore-case

Do not ignore case distinctions in patterns and input data. This is the default. This option is useful for passing to shell scripts that already use -i, to cancel its effects

-v, --invert-match

Invert the sense of matching, to select non-matching lines.

-w, --word-regexp

Select only those lines containing matches that form whole words. The test is that the matching substring must either be at the beginning of the line, or preceded by a non-word

constituent character. Similarly, it must be either at the end of the line or followed by a non-word constituent character. Word-constituent characters are letters, digits, and the

underscore. This option has no effect if -x is also specified.

-x, --line-regexp

Select only those matches that exactly match the whole line. For a regular expression pattern, this is like parenthesizing the pattern and then surrounding it with ^ and $.

-y Obsolete synonym for -i.

General Output Control

-c, --count

Suppress normal output; instead print a count of matching lines for each input file. With the -v, --invert-match option (see below), count non-matching lines.

--color[=WHEN], --colour[=WHEN]

Surround the matched (non-empty) strings, matching lines, context lines, file names, line numbers, byte offsets, and separators (for fields and groups of context lines) with escape

sequences to display them in color on the terminal. The colors are defined by the environment variable GREP\_COLORS. The deprecated environment variable GREP\_COLOR is still

supported, but its setting does not have priority. WHEN is never, always, or auto.

-L, --files-without-match

Suppress normal output; instead print the name of each input file from which no output would normally have been printed. The scanning will stop on the first match.

-l, --files-with-matches

Suppress normal output; instead print the name of each input file from which output would normally have been printed. The scanning will stop on the first match.

-m NUM, --max-count=NUM

Stop reading a file after NUM matching lines. If the input is standard input from a regular file, and NUM matching lines are output, grep ensures that the standard input is

positioned to just after the last matching line before exiting, regardless of the presence of trailing context lines. This enables a calling process to resume a search. When grep

stops after NUM matching lines, it outputs any trailing context lines. When the -c or --count option is also used, grep does not output a count greater than NUM. When the -v or

--invert-match option is also used, grep stops after outputting NUM non-matching lines.

-o, --only-matching

Print only the matched (non-empty) parts of a matching line, with each such part on a separate output line.

-q, --quiet, --silent

Quiet; do not write anything to standard output. Exit immediately with zero status if any match is found, even if an error was detected. Also see the -s or --no-messages option.

-s, --no-messages

Suppress error messages about nonexistent or unreadable files.

**Output :**

varadmash@varadmash-G3-3590:~$ grep "ui" new\_file.txt

Quick

varadmash@varadmash-G3-3590:~$ grep -v "ui" new\_file.txt

The

Brown

Fox

Jumped

Over

The

Fence

**8) sed**

**Manual**

NAME

sed - stream editor for filtering and transforming text

SYNOPSIS

sed [OPTION]... {script-only-if-no-other-script} [input-file]...

DESCRIPTION

Sed is a stream editor. A stream editor is used to perform basic text transformations on an input stream (a file or input from a pipeline). While in some ways similar to an editor which

permits scripted edits (such as ed), sed works by making only one pass over the input(s), and is consequently more efficient. But it is sed's ability to filter text in a pipeline which par‐

ticularly distinguishes it from other types of editors.

-n, --quiet, --silent

suppress automatic printing of pattern space

--debug

annotate program execution

-e script, --expression=script

add the script to the commands to be executed

-f script-file, --file=script-file

add the contents of script-file to the commands to be executed

--follow-symlinks

follow symlinks when processing in place

-i[SUFFIX], --in-place[=SUFFIX]

edit files in place (makes backup if SUFFIX supplied)

-l N, --line-length=N

specify the desired line-wrap length for the `l' command

--posix

disable all GNU extensions.

-E, -r, --regexp-extended

use extended regular expressions in the script (for portability use POSIX -E).

-s, --separate

consider files as separate rather than as a single, continuous long stream.

--sandbox

operate in sandbox mode (disable e/r/w commands).

-u, --unbuffered

load minimal amounts of data from the input files and flush the output buffers more often

-z, --null-data

separate lines by NUL characters

--help

display this help and exit

--version

output version information and exit

If no -e, --expression, -f, or --file option is given, then the first non-option argument is taken as the sed script to interpret. All remaining arguments are names of input files; if no

input files are specified, then the standard input is read.

GNU sed home page: <https://www.gnu.org/software/sed/>. General help using GNU software: <https://www.gnu.org/gethelp/>. E-mail bug reports to: <[bug-sed@gnu.org](mailto:bug-sed@gnu.org)>.

**Output :**

varadmash@varadmash-G3-3590:~$ cat new\_file.txt

The

Quick

Brown

Fox

Jumped

Over

The

Fence

varadmash@varadmash-G3-3590:~$ sed -i "s/Quick/Slow/g" new\_file.txt

varadmash@varadmash-G3-3590:~$ cat new\_file.txt

The

Slow

Brown

Fox

Jumped

Over

The

Fence

**9) chmod**

**Manual**

AME

chmod - change file mode bits

SYNOPSIS

chmod [OPTION]... MODE[,MODE]... FILE...

chmod [OPTION]... OCTAL-MODE FILE...

chmod [OPTION]... --reference=RFILE FILE...

DESCRIPTION

This manual page documents the GNU version of chmod. chmod changes the file mode bits of each given file according to mode, which can be either a symbolic representation of changes to make,

or an octal number representing the bit pattern for the new mode bits.

The format of a symbolic mode is [ugoa...][[-+=][perms...]...], where perms is either zero or more letters from the set rwxXst, or a single letter from the set ugo. Multiple symbolic modes

can be given, separated by commas.

A combination of the letters ugoa controls which users' access to the file will be changed: the user who owns it (u), other users in the file's group (g), other users not in the file's group

(o), or all users (a). If none of these are given, the effect is as if (a) were given, but bits that are set in the umask are not affected.

The operator + causes the selected file mode bits to be added to the existing file mode bits of each file; - causes them to be removed; and = causes them to be added and causes unmentioned

bits to be removed except that a directory's unmentioned set user and group ID bits are not affected.

The letters rwxXst select file mode bits for the affected users: read (r), write (w), execute (or search for directories) (x), execute/search only if the file is a directory or already has

execute permission for some user (X), set user or group ID on execution (s), restricted deletion flag or sticky bit (t). Instead of one or more of these letters, you can specify exactly one

of the letters ugo: the permissions granted to the user who owns the file (u), the permissions granted to other users who are members of the file's group (g), and the permissions granted to

users that are in neither of the two preceding categories (o).

A numeric mode is from one to four octal digits (0-7), derived by adding up the bits with values 4, 2, and 1. Omitted digits are assumed to be leading zeros. The first digit selects the

set user ID (4) and set group ID (2) and restricted deletion or sticky (1) attributes. The second digit selects permissions for the user who owns the file: read (4), write (2), and execute

(1); the third selects permissions for other users in the file's group, with the same values; and the fourth for other users not in the file's group, with the same values.

chmod never changes the permissions of symbolic links; the chmod system call cannot change their permissions. This is not a problem since the permissions of symbolic links are never used.

However, for each symbolic link listed on the command line, chmod changes the permissions of the pointed-to file. In contrast, chmod ignores symbolic links encountered during recursive di‐

rectory traversals.

SETUID AND SETGID BITS

chmod clears the set-group-ID bit of a regular file if the file's group ID does not match the user's effective group ID or one of the user's supplementary group IDs, unless the user has ap‐

propriate privileges. Additional restrictions may cause the set-user-ID and set-group-ID bits of MODE or RFILE to be ignored. This behavior depends on the policy and functionality of the

underlying chmod system call. When in doubt, check the underlying system behavior.

For directories chmod preserves set-user-ID and set-group-ID bits unless you explicitly specify otherwise. You can set or clear the bits with symbolic modes like u+s and g-s. To clear

these bits for directories with a numeric mode requires an additional leading zero, or leading = like 00755 , or =755

RESTRICTED DELETION FLAG OR STICKY BIT

The restricted deletion flag or sticky bit is a single bit, whose interpretation depends on the file type. For directories, it prevents unprivileged users from removing or renaming a file

in the directory unless they own the file or the directory; this is called the restricted deletion flag for the directory, and is commonly found on world-writable directories like /tmp. For

regular files on some older systems, the bit saves the program's text image on the swap device so it will load more quickly when run; this is called the sticky bit.

OPTIONS

Change the mode of each FILE to MODE. With --reference, change the mode of each FILE to that of RFILE.

-c, --changes

like verbose but report only when a change is made

-f, --silent, --quiet

suppress most error messages

-v, --verbose

output a diagnostic for every file processed

--no-preserve-root

do not treat '/' specially (the default)

--preserve-root

fail to operate recursively on '/'

--reference=RFILE

use RFILE's mode instead of MODE values

-R, --recursive

change files and directories recursively

--help display this help and exit

--version

output version information and exit

Each MODE is of the form '[ugoa]\*([-+=]([rwxXst]\*|[ugo]))+|[-+=][0-7]+'.

**Output :**

varadmash@varadmash-G3-3590:~$ ls -l | grep "new\_file.txt"

-rw-rw-r-- 1 varadmash varadmash 43 Sep 4 10:47 new\_file.txt

varadmash@varadmash-G3-3590:~$ chmod 765 new\_file.txt

varadmash@varadmash-G3-3590:~$ ls -l | grep "new\_file.txt"

-rwxrw-r-x 1 varadmash varadmash 43 Sep 4 10:47 new\_file.txt

**10) Loops**

The following types of Loops are found in Linux Shell

1. while do done: Perform a set of commands while a test is true.
2. until do done: Perform a set of commands until a test is true.
3. for do done: Perform a set of commands for each item in a list.
4. select do done: Display a simple menu system for selecting items from a list.

**a) while do done**

Output :

varadmash@varadmash-G3-3590:~$ while [ $count -gt 0 ]

> do

> echo "$count"

> ((count--))

> done

10

9

8

7

6

5

4

3

2

1

**b) Until Do Done**

**Output :**

varadmash@varadmash-G3-3590:~$ counter=1

varadmash@varadmash-G3-3590:~$ until [ $counter -gt 10 ]

> do

> echo $counter

> ((counter++))

> done

1

2

3

4

5

6

7

8

9

10

**c) For Do Done**

**Output :**

varadmash@varadmash-G3-3590:~$ cars="BMW AUDI MERCEDES LAMBORGHINI"

varadmash@varadmash-G3-3590:~$ for car in $cars

> do

> echo "$car"

> done

BMW

AUDI

MERCEDES

LAMBORGHINI

**d) Select Do Done**

**Output :**

varadmash@varadmash-G3-3590:~$ cars="BMW AUDI MERCEDES LAMBORGHINI"

varadmash@varadmash-G3-3590:~$ select car in $cars

> do

> echo "$car"

> done

1) BMW

2) AUDI

3) MERCEDES

4) LAMBORGHINI

#? 1

BMW

#? 2

AUDI

#? 3

MERCEDES

#? 4

LAMBORGHINI

#? ^C

**11) Conditional Loops**

**a) if .. then .. elif .. fi**

**Output :**

varadmash@varadmash-G3-3590:~$ read percentage

85

varadmash@varadmash-G3-3590:~$ if [ $percentage -ge 90 ]

> then

> echo "Grade A"

> elif [ $percentage -ge 80 ]

> then

> echo "Grade B"

> else

> echo "Grade C"

> fi

Grade B

**b) case .. esac**

**Output :**

varadmash@varadmash-G3-3590:~$ read def\_wing

2

varadmash@varadmash-G3-3590:~$ case $def\_wing in

> 1) echo "ARMY";;

> 2) echo "NAVY";;

> 3) echo "AIR FORCE";;

> \*) echo "INVALID";;

> esac

NAVY

**Conclusion :**

1. Basic Linux Commands were studied.

2. Implementation of commands with options done through terminal.