Linux Commands

1. pwd : print name of current working directory.
    a. Synopsis: pwd [option]
    b. -L, --logical : use PWD from environment, even if contains symlinks.
    c. -P , --physical: avoid all symlinks.
    d. Example
        i. INPUT: pwd
        ii. OUTPUT: /home/mehakjotsingh
    e. Windows : cd
2. cd : change the working directory.
    a. Synopsis: cd [-L|-P] [Directory]
    b. cd .. : return to parent directory
    c. -L : force symbolic links to be followed\
    d. -P: use the physical directory structure without using the symlinks
    e. Example:
        i. INPUT: $ cd Desktop
        ii. OUTPUT: /Desktop$
    f. cd
3. cp : copy SOURCE to DEST or multiple sources to DIRECTORY
    a. Synopsis:
        i. cp [OPTION]... SOURCE DEST
        ii. CP [OPTION]... SOURCE.. DIRECTORY
    b. Options:
        i. --attributes-only: don't copy the file data just the attributes
        ii. -f : if an existing destination file cannot be opened, remove it and try again
        iii. -s :make symbolic links instead of copying
    c. Example:
        i. INPUT: cp hello.txt bie.txt
        ii. OUTPUT: (no output, a new file name is created named bie.txt, with same content as hello.txt, there was no bie.txt earlier)
    d. Windows: copy
4. ls: list directory contents
    a. Synopsis: ls [OPTION]... [FILE]...
    b. Options:
        i. -a : Do not ignore entries starting with .
        ii. -l : use long listing format
        iii. -i: print the index number of each file
        iv. -m: fill width with a comma separated list of entries
    c. Example:
        i. INPUT: ls -m
        ii. OUTPUT: bie, hello

    d. Windows: dir

5. mkdir: create DIRECTORY(ies) if they do not exist
    a. Synopsis: mkdir [OPTION]... DIRECTORY…
    b. Options:
        i. -p: make parent directories as needed\
        ii. -m : =MODE (rwx), set file mode
    c. Example:
        i. INPUT: mkdir -p -v a/b
        ii. OUTPUT: mkdir: created directory 'a'
mkdir: created directory 'a/b
    d. Windows: mkdir
6. rm: remove files or directories (directories are not removed by default)
    a. Synopsis: rm [OPTION]... [FILE]...
    b. Options:
        i. -d : remove empty directories
        ii. -f, --force: ignore non existence files and arguments, do not prompt
        iii. -i : prompt before every removal
        iv. -r,-R : to remove each listed directory with all of its content (recursively)
    c. Examples:
        i. INPUT: rm -v -r a
        ii. OUTPUT: removed directory ‘a/b’
removed directory ‘a’
    d. Windows: del
7. chmod: change file mode bits
    a. Synopsis:
        i. chmod [OPTION]... MODE[,MODE]... FILE...
        ii. chmod [OPTION]... OCTAL-MODE... FILE…
    b. Options:
        i. -f,-silent : suppress most error messages
        ii. -R : change files and directories recursively
        iii. -c, --changes: reports when a change is made
    c. Examples:
        i. INPUT: chmod -c u=r hello.txt
        ii. OUTPUT:mode of 'hello.txt' changed from 0644 (rw-r--r--) to 0444 (r--r--r--)
    d. Windows: attrib
8. gzip: compress or expand files
    a. Synopsis: gzip [OPTIONS]... [name]..
    b. Options:
        i. -d, --decompress, --uncompress: Decompress
        ii. -k : keep input files during decompression or compression

        iii. -l, --list: for each compressed file display the compressed size, uncompressed size, ratio, uncompressed name.
    c. Example:
        i. INPUT: gzip -v hello.txt
        ii. OUTPUT: hello: 0.0% -- replaced with hello.gz
    d. Windows: ~no default command

9. find : search for files in a directory hierarchy
    a. Synopsis: find [OPTIONS]... [starting-point]... [expression]
    b. Options:
        i. -P: never follow symbolic links(default)
        ii. -L: always follow symbolic links
        iii. -H: Do not follow symbolic links, except while processing the command line arguments.
        iv. -D : debugoptions, Print diagnostic information
    c. Example:
        i. INPUT: find abc
        ii. OUTPUT:

```
abc
abc/a
```

    d. Windows: find

10. less: less command does not read the entire input, like to read contents of text file one page(one screen) per time.
    a. Synopsis: less filename
    b. Example:
        i. Input: less hello.txt
        ii. Output: (output screen printing contents of hello.txt)
    c. Windows: more

11. tail: output the last part of files, print the last 10 lines of each file to standard output.
    a. Synopsis: tail [OPTION]... [FILE]...
    b. Options:
        i. -c,--bytes=NUM: output the last NUM bytes;
        ii. -n,--lines=NUM: output the last NUM lines, instead of last 10;
        iii. -q: never output headers giving file names
    c. Example:
        i. Input: tail hello.txt
        ii. Output: (last 10 lines of hello.txt)
    d. Windows: tail

12. top: display linux process, system summary operations
    a. Example:
        i. Input: top t
        ii. Output:

```
top- 19:06:52 up 6 days,  2:03,  1 user,  load average: 0.40, 0.78, 0.74
Tasks: 344 total,   1 running, 273 sleeping,   0 stopped,   0 zombie
%Cpu(s):  6.2 us,  1.5 sy,  0.0 ni, 91.6 id,  0.4 wa,  0.0 hi,  0.2 si,  0.0 st
```

```
KiB Mem :  8062752 total,   230488 free,  6137412 used,  1694852 buff/cache
KiB Swap:  2097148 total,  1993140 free,   104008 used.   657032 avail Mem
```

   b. Windows: mem

13. wc: print newline, word, and byte counts for each file
   a. Synopsis: wc [OPTION]... [FILE]...
   b. Option:
      i. -c: print the byte counts
      ii. -m: print the character counts
      iii. -l: print the newline counts
   c. Example:
      i. Input: wc hello.txt
      ii. Output: 12 11 25 hello.txt
   d. Windows: fc
14. diff: compare the files line by line
   a. Synopsis: diff [OPtiON]... FILES
   b. Option:
      i. -q: report only when files differ
      ii. -y: output in 2 columns
      iii. -s:report when 2 files are same
   c. Example:
      i. Input: diff hello.txt hello2.txt
      ii. Output: Files hello.txt and hello2.txt are identical
   d. Windows: fc
15. kill: send a signal to a process, default is TERM.
   a. Synopsis: kill [options] <pid> [...]
   b. Options:
      i. -l : to list the available signals
      ii. -L: list signal names in a nice table
   c. Example:
      i. Input: kill -9 -1
      ii. Output: (kill all processes you can kill)
   d. Windows: task kill
16. Ifconfig: configure network interface
   a. Synopsis: ifconfig [-v] [-a][-s] [interface]
   b. Option:
      i. -a: display all interfaces which are currently available, even if down
      ii. -s: display a a short list
      iii. -down: causes the driver for this interface to be shut down.
   c. Example:
      i. Input: ifconfig
      ii. Output: enp1s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500

```
ether 50:9a:4c:d1:43:e4  txqueuelen 1000  (Ethernet)
RX packets 0  bytes 0 (0.0 B)
RX errors 0  dropped 0  overruns 0  frame 0
TX packets 0  bytes 0 (0.0 B)
TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

   d. Windows: ipconfig

17. shutdown: halt, power-off or reboot the machine
   a. Synopsis: shutdown [OPTIONS…] [TIME] [WALL...]
   b. Options:
      i. -P: poweroff
      ii. -r: reboot
      iii. -k : do not halt or power off or reboot , just write wall messages
   c. Example: shutdown -P
   d. Windows: shutdown
18. which: locate a command
   a. Synopsis: which [-a] filename …
   b. Options:
      i. -a: print all matching pathnames of each argument
   c. Example:
      i. Input: which python
      ii. Output: /usr/bin/python
   d. Windows: where
19. cat: concatenate files and print on the standard output
   a. Synopsis: cat [OPTION]... [FILE]...
   b. Options:
      i. -b: number non empty output lines
      ii. -E: display $ at end of each line
      iii. -s
   c. Example:
      i. Input: cat hello.txt hello2.txt
      ii. Output: (concatenated text of both files)
   d. Windows: type
20. mv: move files
   a. Synopsis: mv [OPTIONS]... [-T] SOURCE DEST
   b. Options:
      i. -b : make a backup of each existing destination file
      ii. -n: do not overwrite existing file
      iii. -f: do not prompt before over writting
   c. Example:
      i. Input: mv -v hello.txt a/hello.txt
      ii. Output: renamed ‘hello.txt’ -> ‘a/hello.txt’
   d. Windows: move
21. grep: searches for PATTERN in each FILE.
   a. Synopsis: grep [OPTIONS] PATTERN [FILE...]
   b. Options:

- i. -E,--extended-regexp: interpret PATTERN as an extended regular expression
- ii. -F, --fixed-strings:Interpret PATTERN as a list of fixed strings
- iii. -G: Interpret PATTERN as a basic regular expression
- iv. -c: Display count of matching lines for each input file.

- c. Example:
    - i. Input: grep -c 123 hello2.txt
    - ii. Output: 0
- d. Windows: find

22. cut: remove sections from each line of files, print selected part if lines from each FILE to standard output.
    - a. Synopsis: cut OPTION… [FILE]...
    - b. Options:
        - i. -c=LIST : select only these characters
        - ii. -b=LIST : select only these bytese
        - iii. -d=DELIM: use DELIM instead of TAB for field delimiter
    - c. Example:
        - i. Input: cut -c 1 hello2.txt
        - ii. Output: (prints the first column of characters)
    - d. Windows:cut
23. history: used to view previous executed command
    - a. Synopsis: history
    - b. Options:
        - i. [n]: To show the limited number of commands that executed previously
        - ii. history | grep PATTERN: to find PATTERN in history
    - c. Example:
        - i. Input: history | grep g++
        - ii. Output:

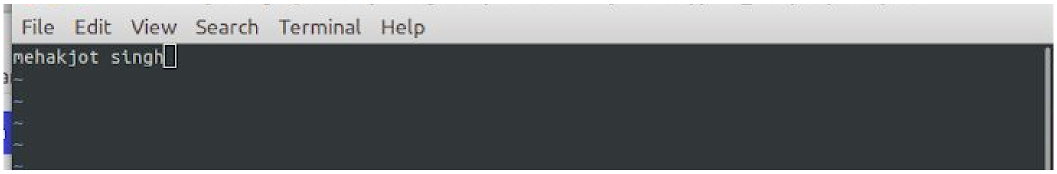
```
1031  g++ sumOfmedians.cpp
1032  g++ bitonicSearch.cpp
1034  g++ test.cpp4
```

    - d. Windows: F7
24. echo: display a line of text
    - a. Synopsis: echo [SHORT-OPTION].... [STRING]....
    - b. Options:
        - i. -n: do not output trailing newline
        - ii. -e: enable interpretation of backslash escapes
    - c. Example:
        - i. Input: echo hi
        - ii. Output: hi
    - d. Windows: echo

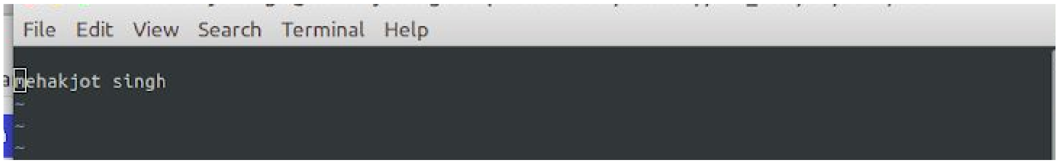
Source:$ man [command] (linux manual)

Vi Editor

1. Starting Vi Editor: vi filename
2. Command Mode : Understands only commands, allows to move the cursor, copy, cut paste text.
3. Insert Mode : for inserting text, we can enter this mode by pressing ‘i’ in command mode and exit by pressing ‘Esc’. Once in insert mode any key is understood as input for the file.
4. Editing in vi:
    a. i :Insert at cursor **(goes into insert mode)**
    b. a : Write after cursor **(goes into insert mode)**
    c. A:: Write at the end of line **(goes into insert mode)**
    d. ESC :Terminate insert mode
    e. u : Undo last change
    f. U : Undo all changes to the entire line
    g. o: Open a new line **(goes into insert mode)**
    h. dd: Delete line
    i. ndd : Delete n lines
5. Moving cursor
    a. k - Move cursor up
    b. j - Move cursor down
    c. h -Move cursor left
    d. l - Move cursor right
6. Copying ,Cutting, Pasting
    a. y: copies the current character or the selected text
    b. d: copies the current character or the selected text
    c. p: paste the copied text after cursor



```
mehakjot singh
~
~
~
~
~
```



```
mehakjot singh
~
~
~
```

Steps Taken

1. Press i to enter insert mode
2. Enter text : “mehakjo singH”
3. Press Escape to exit insert mode
4. Press ctrl V to select line
5. Press d to selected text

6. Press p to paste in next line