Linux Commands

COM301-P Exercise 1

G S Santhosh Raghul
COE18B045

Question 1

1. Test drive and understand the usage of all the commands given in the 50 Most Frequently Used UNIX / Linux Commands and linuxcommands.pdf

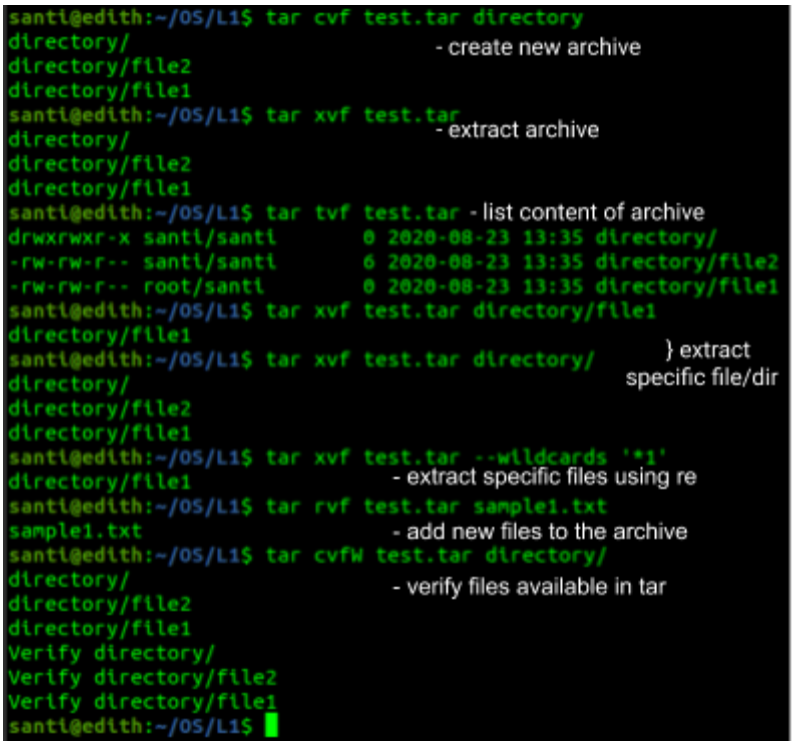
1. tar command

`tar c<other_options> tarfilename dirname` : create a new tar archive with contents of dirname

`tar x<other_options> tarfilename` : extract contents of the tar archive in the current working directory

Options:

- c – create a new archive
- x – extract files from archive
- v – verbosely list files which are processed.
- f – following is the archive file name
- z – filter the archive through gzip
- j – filter the archive through bzip2



2. grep command

`grep -i "text" filename` : search for a given string in a file
`grep -i "text" filename` : search for a given string in a file (case in-sensitive)
`grep -B 2 "text" filename` : print the matched line, along with the 3 lines after it
`grep "text" re` : search for a given string in all files that match the

```
santi@edith:~/OS/L1$ grep "UNIX" sample1.txt
UNIX is a free OS.
UNIX is a free OS.
UNIX is a free OS.
santi@edith:~/OS/L1$ grep -i "UNIX" sample1.txt
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages syste
m and process activities.
Unix is a great OS.
UNIX is a free OS.
UNixOS systems use a centralized operating system kernel which manages sys
tem and process activities.
A user can also run multiple programs at the same time; hence Unix is a mu
ltitasking environment.
UNIX is a free OS.
santi@edith:~/OS/L1$ grep -B 2 "powerful" sample2.txt
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
santi@edith:~/OS/L1$ grep "Multiuser" sample*
sample1.txt:Multiuser operating system.
sample2.txt:Multiuser operating system.
santi@edith:~/OS/L1$
```

3. find command

`find text` : displays full path of files, directories **from the current directory** which match the given text

```
santi@edith:~/OS$ find ./*/sample*
./L1/sample1.txt
./L1/sample2.txt
santi@edith:~/OS$ find L1/*.pdf
L1/50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf
L1/Exercise-I LINUX COMMANDS.pdf
L1/linuxcommands.pdf
santi@edith:~/OS$
```

4. ssh command

`ssh user@ip_address` or `ssh -l user remotehost.example.com` : connect to a host through ssh

`ssh -v -l user remotehost.example.com` : debug ssh client

```
santi@edith:~$ ssh santi@localhost
santi@localhost's password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-42-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

 * Are you ready for Kubernetes 1.19? It's nearly here! Try RC3 with
   sudo snap install microk8s --channel=1.19/candidate --classic

   https://microk8s.io/ has docs and details.

0 updates can be installed immediately.
0 of these updates are security updates.

Your Hardware Enablement Stack (HWE) is supported until April 2025.
*** System restart required ***
Last login: Wed Aug 26 17:21:28 2020 from 127.0.0.1
santi@edith:~$ ssh -V
OpenSSH_8.2p1 Ubuntu-4ubuntu0.1, OpenSSL 1.1.1f  31 Mar 2020
santi@edith:~$
```

5. sed command

`sed -n '1!G;h;$p' file` : print file in reverse order

`sed '/./=' file | sed 'N; s/\n/ /'` : add line numbers in a file

```
santi@edith:~/OS/L1$ sed -n '1!G;h;$p' test.txt
santhosh
is
name
my
santi@edith:~/OS/L1$ sed '/./=' test.txt | sed 'N; s/\n/ /'
1 my
2 name
3 is
4 santhosh
santi@edith:~/OS/L1$
```

6. awk command

`awk '!($0 in array) { array[$0]; print }' file` : remove duplicate lines and print the file content

`awk '{print $2,$5;}' file` : print only specific field from a file

```
santi@edith:~/OS/L1$ cat test.txt
apple
orange
grape
apple
santi@edith:~/OS/L1$ awk '!($0 in array) { array[$0]; print }' test.txt
apple
orange
grape
santi@edith:~/OS/L1$ awk '{print $2,$5;}' sample2.txt
was in
are variants
people a
user run
is OS.
operating
another

santi@edith:~/OS/L1$
```

7. vim command

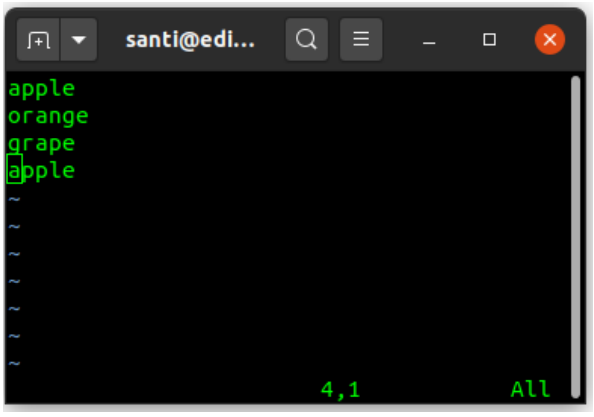
`vim file` : open file in vim

`vim +5 file` : open file in vim and go to line number 5

`vim +/text file` : open file in vim and go to the first occurrence of the specified text

`vim -R file` : open file in vim in read only mode

```
santi@edith:~/OS/L1$ vim test.txt
santi@edith:~/OS/L1$ vim +2 test.txt
santi@edith:~/OS/L1$ vim +/grape test.txt
santi@edith:~/OS/L1$ vim -R test.txt
```



8. diff command

`diff file1 file2` : compare the 2 files

`diff -w file1 file2` : ignore white space while comparing

```
santi@edith:~/OS/L1$ cat test.txt
apple
orange
grape
apple
santi@edith:~/OS/L1$ cat test2.txt
apple
grape
orange
apple
santi@edith:~/OS/L1$ diff test.txt test2.txt
2d1
< orange
3a3
> orange
santi@edith:~/OS/L1$
```

9. sort command

`sort filename` : sorts and displays the contents of the given file in ascending order

`sort -r filename` : descending order

`sort -t: -k 3n file | more` : sort file by 3rd field

```
santi@edith:~/OS/L1$ sort sample1.txt
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, mem
ory management, process management, handling input and output, and controlling p
eripheral devices such as disk drives and printers.
A user can also run multiple programs at the same time; hence Unix is a multitas
king environment.
Multiuser operating system.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
This is a test document.
UNIX is a free OS.
UNIX is a free OS.
UNIX is a free OS.
Unix is a great OS.
Unix is a great OS.
UNixOS systems use a centralized operating system kernel which manages system an
d process activities.
Unix systems use a centralized operating system kernel which manages system and
process activities.
Yet another powerful OS.
santi@edith:~/OS/L1$
```

10. export command

`export -p` : prints all the variables

`export <variable_name>=<value>` : This command exports the variable with its corresponding value.

```
santi@edith:~$ export -p
declare -x CLUTTER_IM_MODULE="ibus"
declare -x COLORTERM="truecolor"
declare -x DBUS_SESSION_BUS_ADDRESS="unix:path=/run/user/1000/bus"
declare -x DESKTOP_SESSION="ubuntu"
declare -x DISPLAY=":1"
declare -x GDMSESSION="ubuntu"
declare -x GNOME_DESKTOP_SESSION_ID="this-is-deprecated"
declare -x GNOME_SHELL_SESSION_MODE="ubuntu"
declare -x GNOME_TERMINAL_SCREEN="/org/gnome/Terminal/screen/2b9d5ad7_4755_4557_
a624_c089412a5492"
declare -x GNOME_TERMINAL_SERVICE=":1.646"
declare -x GPG_AGENT_INFO="/run/user/1000/gnupg/S.gpg-agent:0:1"
declare -x GTK_IM_MODULE="ibus"
declare -x GTK_MODULES="gail:atk-bridge"
declare -x HOME="/home/santi"
```

11. xargs command

`ls | xargs -n1 -i cp {} dup` : This command copies all the files to another folder using xargs

```
santi@edith:~/OS/L1$ ls dup/
santi@edith:~/OS/L1$ ls | xargs -n1 -i cp {} dup
cp: -r not specified; omitting directory 'dircopy'
cp: -r not specified; omitting directory 'directory'
cp: -r not specified; omitting directory 'dup'
cp: -r not specified; omitting directory 'empty-dir'
santi@edith:~/OS/L1$ ls dup/
'50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf'   sample1.txt
'Exercise-I LINUX COMMANDS.pdf'                                    sample2.txt
 Input.txt                                                         sample.txt
 linuxcommands.pdf                                                 test.txt
 newfile.txt
santi@edith:~/OS/L1$
```

12. ls command

`ls` : (short for list) displays the contents of the current directory

`ls -l` : -l option displays the content in a long listing format which contains user permissions and ownership of the files.

`ls -R` : -R (stands for recursive) option in ls displays the files in current directory and the subdirectories.

`ls -a` : lists all files including hidden files/dirs. Hidden files/dirs start with a '.'

```
santi@edith:~/OS/L1$ ls
'50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf'
 dircopy
 directory
 empty-dir
'Exercise-I LINUX COMMANDS.pdf'
 Input.txt
 linuxcommands.pdf
 newfile.txt
 sample1.txt
 sample2.txt
 sample.txt
 test.txt
santi@edith:~/OS/L1$ ls -l
total 1336
-rw-rw-r-- 1 santi santi 735087 Aug 21 20:55 '50_Most_Frequently_Used_UNIX _ Linux Commands_With
Examples.pdf'
drwxrwxr-x 2 santi santi   4096 Aug 27 20:32  dircopy
drwxrwxr-x 2 santi santi   4096 Aug 26 09:03  directory
drwxrwxr-x 2 santi santi   4096 Aug 23 13:34  empty-dir
-rw-rw-r-- 1 santi santi 120472 Aug 21 20:53 'Exercise-I LINUX COMMANDS.pdf'
-rw-rw-r-- 1 santi santi   1484 Aug 27 20:07  Input.txt
-rw-rw-r-- 1 santi santi 469423 Aug 21 20:55  linuxcommands.pdf
-rw-rw-r-- 1 santi santi    591 Aug 27 20:30  newfile.txt
-rw-rw-r-- 1 santi santi    893 Aug 23 15:30  sample1.txt
-rw-rw-r-- 1 santi santi    591 Aug 23 15:16  sample2.txt
-rw-rw-r-- 1 santi santi    591 Aug 27 19:54  sample.txt
-rw-rw-r-- 1 santi santi   1484 Aug 27 20:34  test.txt
santi@edith:~/OS/L1$
```

```
santi@edith:~/OS/L1$ ls -R
.:
'50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf'   linuxcommands.pdf
 dircopy                                                           newfile.txt
 directory                                                         sample1.txt
 empty-dir                                                         sample2.txt
'Exercise-I LINUX COMMANDS.pdf'                                    sample.txt
 Input.txt                                                         test.txt

./dircopy:
file1  file2

./directory:
file1  file2

./empty-dir:
santi@edith:~/OS/L1$ ls -a
 .                                                                 Input.txt
 ..                                                                linuxcommands.pdf
'50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf'   newfile.txt
 dircopy                                                           sample1.txt
 directory                                                         sample2.txt
 empty-dir                                                         sample.txt
'Exercise-I LINUX COMMANDS.pdf'                                    test.txt
```

13. pwd command

`pwd` : short for print working directory. prints the path of the current directory

14. cd command

`cd` : without any input directory, the directory is changed to the home directory
`cd directory` : changes to the specified directory
`cd ..` : changes to the parent directory of the current working directory
`cd -` : toggles between last used directory

```
santi@edith:~/OS/L1$ pwd
/home/santi/OS/L1
santi@edith:~/OS/L1$ cd
santi@edith:~$ cd OS/L1
santi@edith:~/OS/L1$ cd ..
santi@edith:~/OS$ cd -
/home/santi/OS/L1
santi@edith:~/OS/L1$
```

15. gzip command

`gzip test.txt` : create a .gz compressed file
`gzip -d test.txt.gz` : uncompress a .gz file
`gzip -l test.txt.gz` : display compression ratio of the compressed file

```
santi@edith:~/OS/L1/directory$ ls
file1  file2  test.txt
santi@edith:~/OS/L1/directory$ gzip test.txt
santi@edith:~/OS/L1/directory$ ls
file1  file2  test.txt.gz
santi@edith:~/OS/L1/directory$ gzip -l test.txt.gz
         compressed        uncompressed  ratio uncompressed_name
                641                1484  58.6% test.txt
santi@edith:~/OS/L1/directory$ gzip -d test.txt.gz
santi@edith:~/OS/L1/directory$ ls
file1  file2  test.txt
```

16. bzip2 command

`bzip2 test.txt` : create a .bz2 compressed file

`bzip -d test.txt.bz2` : uncompress a .bz2 file

```
santi@edith:~/OS/L1/directory$ ls
file1  file2  test.txt
santi@edith:~/OS/L1/directory$ bzip2 test.txt
santi@edith:~/OS/L1/directory$ ls
file1  file2  test.txt.bz2
santi@edith:~/OS/L1/directory$ bzip2 -d test.txt.bz2
santi@edith:~/OS/L1/directory$ ls
file1  file2  test.txt
santi@edith:~/OS/L1/directory$
```

17. unzip command

`unzip filename.zip` : extract the zip file

`unzip -l filename.zip` : view the contents of the zip file without extracting

```
santi@edith:~/OS/L1$ unzip -l directory.zip
Archive:  directory.zip
  Length      Date    Time    Name
---------  ---------- -----   ----
        0  2020-08-28 20:18   directory/
        6  2020-08-23 13:35   directory/file2
        0  2020-08-23 13:35   directory/file1
     1484  2020-08-28 20:17   directory/test.txt
---------                     -------
     1490                     4 files
santi@edith:~/OS/L1$ unzip directory.zip
Archive:  directory.zip
replace directory/file2? [y]es, [n]o, [A]ll, [N]one, [r]ename: A
  inflating: directory/file2
  inflating: directory/file1
  inflating: directory/test.txt
santi@edith:~/OS/L1$
```

18. shutdown command

`shutdown -h now` : shuts the system down immediately

`shutdown -h +10` : shuts the system down in 10 minutes

`shutdown -r now` : reboots the system immediately

`shutdown -Fr now` : force the filesystem check during reboot

```
santi@edith:~/OS/L1$ shutdown -h +2
Shutdown scheduled for Fri 2020-08-28 20:46:40 IST, use 'shutdown -c' to cancel.
santi@edith:~/OS/L1$ shutdown -c
```

19. ftp command

`ftp` stands for file transfer protocol. This command is used to connect to an ftp server.Once successfully connected to the server, mls can be used to list the files on the server, `mget <file>` can be used to download the necessary files.

20. crontab command

`crontab -u user -l` : view crontab entry for a specific user

```
santi@edith:~/OS/L1$ crontab -u santi -l
no crontab for santi
santi@edith:~/OS/L1$
```

21. service command

`service --status-all` : check the status of all the services
`service ssh status` : check status of the ssh service
`service ssh restart` : restart the ssh service

```
santi@edith:~/OS/L1$ service --status-all
 [ + ]  acpid
 [ - ]  alsa-utils
 [ - ]  anacron
 [ + ]  apparmor
 [ + ]  apport
 [ + ]  avahi-daemon
 [ + ]  bluetooth
 [ - ]  console-setup.sh
 [ + ]  cron
 [ + ]  cups
 [ + ]  cups-browsed
 [ + ]  dbus
```

```
santi@edith:~/OS/L1$ service ssh status
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2020-08-28 09:18:56 IST; 11h ago
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 973 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 991 (sshd)
      Tasks: 1 (limit: 9350)
     Memory: 2.4M
     CGroup: /system.slice/ssh.service
             └─991 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

Aug 28 09:18:54 edith systemd[1]: Starting OpenBSD Secure Shell server...
Aug 28 09:18:56 edith sshd[991]: Server listening on 0.0.0.0 port 22.
Aug 28 09:18:56 edith sshd[991]: Server listening on :: port 22.
Aug 28 09:18:56 edith systemd[1]: Started OpenBSD Secure Shell server.
santi@edith:~/OS/L1$ service ssh restart
santi@edith:~/OS/L1$
```

22. ps command

`ps -a` : view all processes

`ps -aN` : view processes not associated to a terminal

`ps -ef` : view current running processes

`ps -efH` : view current running processes in a tree structure

```
santi@edith:~/OS/L1$ ps -a
    PID TTY          TIME CMD
   1055 tty1     00:00:10 Xorg
   1289 tty1     00:00:00 gnome-session-b
   2418 tty2     00:09:53 Xorg
   2491 tty2     00:00:00 gnome-session-b
  18995 pts/0    00:00:00 ps
santi@edith:~/OS/L1$ ps -aN | head -n 6
    PID TTY          TIME CMD
      1 ?        00:00:13 systemd
      2 ?        00:00:00 kthreadd
      3 ?        00:00:00 rcu_gp
      4 ?        00:00:00 rcu_par_gp
      9 ?        00:00:00 mm_percpu_wq
santi@edith:~/OS/L1$ ps -ef | head -n 6
UID          PID    PPID  C STIME TTY          TIME CMD
root           1       0  0 09:18 ?        00:00:13 /sbin/init splash
root           2       0  0 09:18 ?        00:00:00 [kthreadd]
root           3       2  0 09:18 ?        00:00:00 [rcu_gp]
root           4       2  0 09:18 ?        00:00:00 [rcu_par_gp]
root           9       2  0 09:18 ?        00:00:00 [mm_percpu_wq]
santi@edith:~/OS/L1$ ps -efH | head -n 6
UID          PID    PPID  C STIME TTY          TIME CMD
root           2       0  0 09:18 ?        00:00:00 [kthreadd]
root           3       2  0 09:18 ?        00:00:00   [rcu_gp]
root           4       2  0 09:18 ?        00:00:00   [rcu_par_gp]
root           9       2  0 09:18 ?        00:00:00   [mm_percpu_wq]
root          10       2  0 09:18 ?        00:00:00   [ksoftirqd/0]
santi@edith:~/OS/L1$
```

23. free command

`free` : displays free, used, swap memory available in the system in bytes

`free -g` : displays free, used, swap memory available in the system in gb

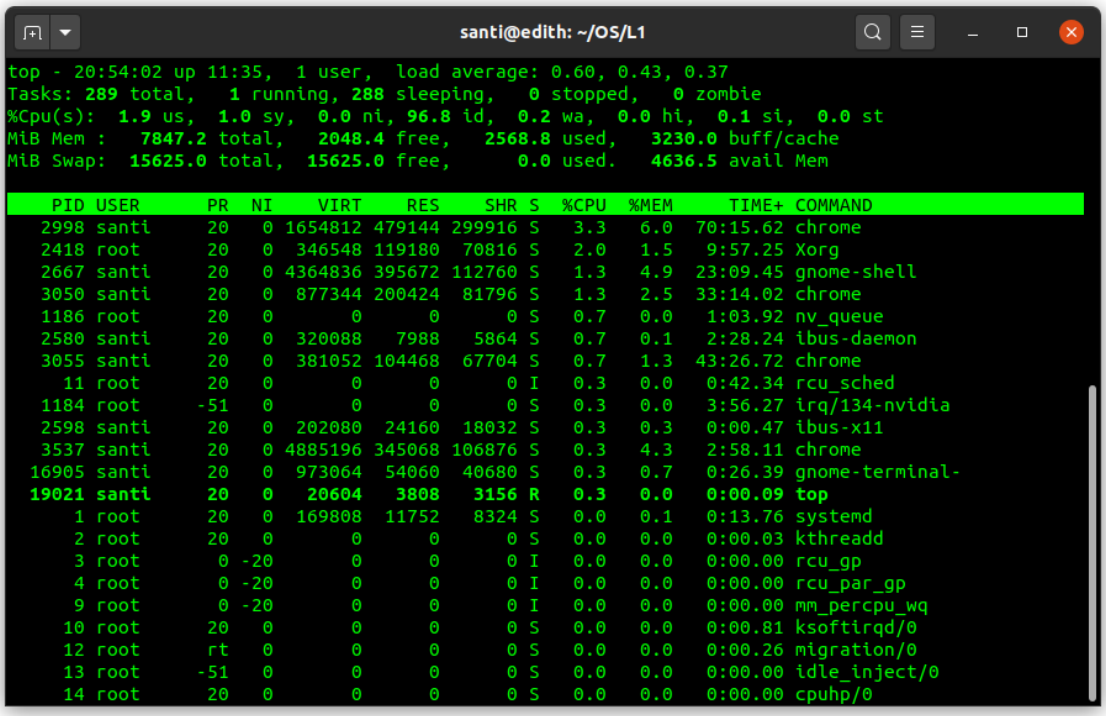
`free -t` : displays free, used, swap memory available in the system along with total memory available

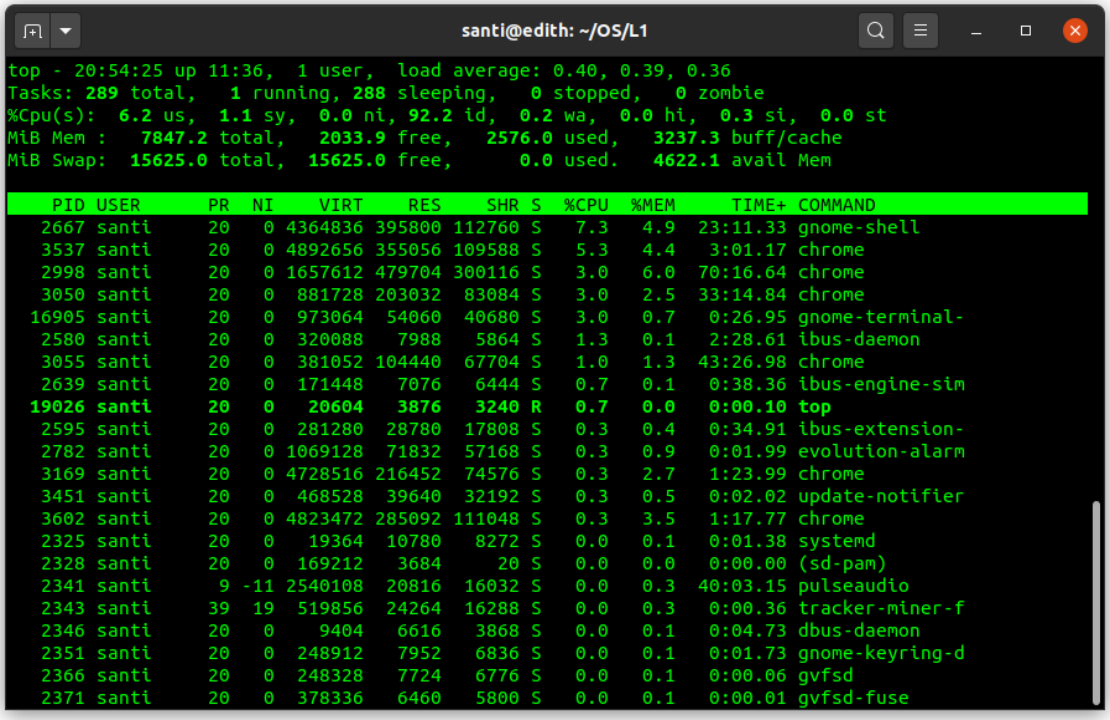
```
santi@edith:~/OS/L1$ free
               total        used        free      shared  buff/cache   available
Mem:         8035544     2633096     2091584      352372     3310864     4741376
Swap:       15999996           0    15999996
santi@edith:~/OS/L1$ free -g
               total        used        free      shared  buff/cache   available
Mem:               7           2           1           0           3           4
Swap:             15           0          15
santi@edith:~/OS/L1$ free -t
               total        used        free      shared  buff/cache   available
Mem:         8035544     2629652     2101500      345880     3304392     4751312
Swap:       15999996           0    15999996
Total:      24035540     2629652    18101496
santi@edith:~/OS/L1$
```

24. top command

`top` : display top processes in the system

`top -u user` : display top processes in the system that belongs to the specified user





25. df command

`df -k` : displays the file system disk space usage in bytes

`df -h` : displays the file system disk space usage in human readable format

`df -T` : displays the file system disk space usage along with type of file system

```
santi@edith:~/OS/L1$ df -k
Filesystem     1K-blocks     Used Available Use% Mounted on
udev             3989620        0   3989620   0% /dev
tmpfs             803556     2044    801512   1% /run
/dev/sda8       39110168  8625452  28468308  24% /
tmpfs            4017772   349532   3668240   9% /dev/shm
tmpfs               5120        4      5116   1% /run/lock
tmpfs            4017772        0   4017772   0% /sys/fs/cgroup
/dev/loop0         56320    56320         0 100% /snap/core18/1880
/dev/loop1         56704    56704         0 100% /snap/core18/1885
/dev/loop4         30720    30720         0 100% /snap/snapd/8790
/dev/loop2         30720    30720         0 100% /snap/snapd/8542
/dev/loop3        297472   297472         0 100% /snap/vlc/1700
/dev/sda9      106397392 94550104   6399560  94% /home
/dev/sda1         262144    75588    186556  29% /boot/efi
tmpfs             803552       28    803524   1% /run/user/125
tmpfs             803552       52    803500   1% /run/user/1000
/dev/mmcblk0p1   1914880       16   1914864   1% /media/santi/E052-86AB
```

```
santi@edith:~/OS/L1$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            3.9G     0  3.9G   0% /dev
tmpfs           785M  2.0M  783M   1% /run
/dev/sda8        38G  8.3G   28G  24% /
tmpfs           3.9G  341M  3.5G   9% /dev/shm
tmpfs           5.0M  4.0K  5.0M   1% /run/lock
tmpfs           3.9G     0  3.9G   0% /sys/fs/cgroup
/dev/loop0       55M   55M     0 100% /snap/core18/1880
/dev/loop1       56M   56M     0 100% /snap/core18/1885
/dev/loop4       30M   30M     0 100% /snap/snapd/8790
/dev/loop2       30M   30M     0 100% /snap/snapd/8542
/dev/loop3      291M  291M     0 100% /snap/vlc/1700
/dev/sda9       102G   91G  6.2G  94% /home
/dev/sda1       256M   74M  183M  29% /boot/efi
tmpfs           785M   28K  785M   1% /run/user/125
tmpfs           785M   52K  785M   1% /run/user/1000
/dev/mmcblk0p1  1.9G   16K  1.9G   1% /media/santi/E052-86AB
```

```
santi@edith:~/OS/L1$ df -T
Filesystem     Type      1K-blocks     Used Available Use% Mounted on
udev           devtmpfs    3989620        0   3989620   0% /dev
tmpfs          tmpfs        803556     2044    801512   1% /run
/dev/sda8      ext4       39110168  8625452  28468308  24% /
tmpfs          tmpfs       4017772   340640   3677132   9% /dev/shm
tmpfs          tmpfs          5120        4      5116   1% /run/lock
tmpfs          tmpfs       4017772        0   4017772   0% /sys/fs/cgroup
/dev/loop0     squashfs      56320    56320         0 100% /snap/core18/1880
/dev/loop1     squashfs      56704    56704         0 100% /snap/core18/1885
/dev/loop4     squashfs      30720    30720         0 100% /snap/snapd/8790
/dev/loop2     squashfs      30720    30720         0 100% /snap/snapd/8542
/dev/loop3     squashfs     297472   297472         0 100% /snap/vlc/1700
/dev/sda9      ext4      106397392 94550084   6399580  94% /home
/dev/sda1      vfat         262144    75588    186556  29% /boot/efi
tmpfs          tmpfs        803552       28    803524   1% /run/user/125
tmpfs          tmpfs        803552       52    803500   1% /run/user/1000
/dev/mmcblk0p1 vfat        1914880       16   1914864   1% /media/santi/E052-86AB
santi@edith:~/OS/L1$
```

26. kill command

`kill -9 <pid>` : kill the specified process

```
santi@edith:~/Desktop$ ./a.out
Killed
santi@edith:~/Desktop$
```

```
santi@edith:~/OS/L1$ ps -a
    PID TTY          TIME CMD
   1055 tty1     00:00:10 Xorg
   1289 tty1     00:00:00 gnome-session-b
   2418 tty2     00:10:09 Xorg
   2491 tty2     00:00:00 gnome-session-b
  19169 pts/1    00:00:00 a.out
  19171 pts/0    00:00:00 ps
santi@edith:~/OS/L1$ kill -9 19169
santi@edith:~/OS/L1$
```

27. rm command

`rm filename` : delete the specified file

`rm -i filename` : delete the specified file, ask for a prompt first

`rm -i file*` : print the filename and get confirmation before removing the file

`rm -r directory` : delete a directory and everything inside it recursively

```
santi@edith:~/OS/L1$ ls
'50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf'   linuxcommands.pdf
 dircopy                                                           newfile.txt
 directory                                                         sample1.txt
 directory.zip                                                     sample2.txt
 empty-dir                                                         sample.txt
'Exercise-I LINUX COMMANDS.pdf'                                    test.txt
 Input.txt
santi@edith:~/OS/L1$ rm test.txt
santi@edith:~/OS/L1$ rm -i newfile.txt
rm: remove regular file 'newfile.txt'? y
santi@edith:~/OS/L1$ rm -r dircopy
santi@edith:~/OS/L1$ ls
'50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf'   Input.txt
 directory                                                         linuxcommands.pdf
 directory.zip                                                     sample1.txt
 empty-dir                                                         sample2.txt
'Exercise-I LINUX COMMANDS.pdf'                                    sample.txt
santi@edith:~/OS/L1$
```

28. cp command

`cp -p file1 file2` : copy file1 to file2 preserving the mode, ownership and timestamp

`cp -i file1 file2` : copy file1 to file2, prompt if a file named file2 already exists

```
santi@edith:~/OS/L1/directory$ ls
file1  file2  test.txt
santi@edith:~/OS/L1/directory$ cp -p test.txt new.txt
santi@edith:~/OS/L1/directory$ ls
file1  file2  new.txt  test.txt
santi@edith:~/OS/L1/directory$ cp -i test.txt new.txt
cp: overwrite 'new.txt'? y
santi@edith:~/OS/L1/directory$
```

29. mv command

`mv -i file1 file2` : rename file1 to file2, prompt if a file named file2 already exists

`mv -f file1 file2` : rename file1 to file2, replace the old file if a file named file2 already exists

`mv -v file1 file2` : -v will print what is happening when the file is being renamed

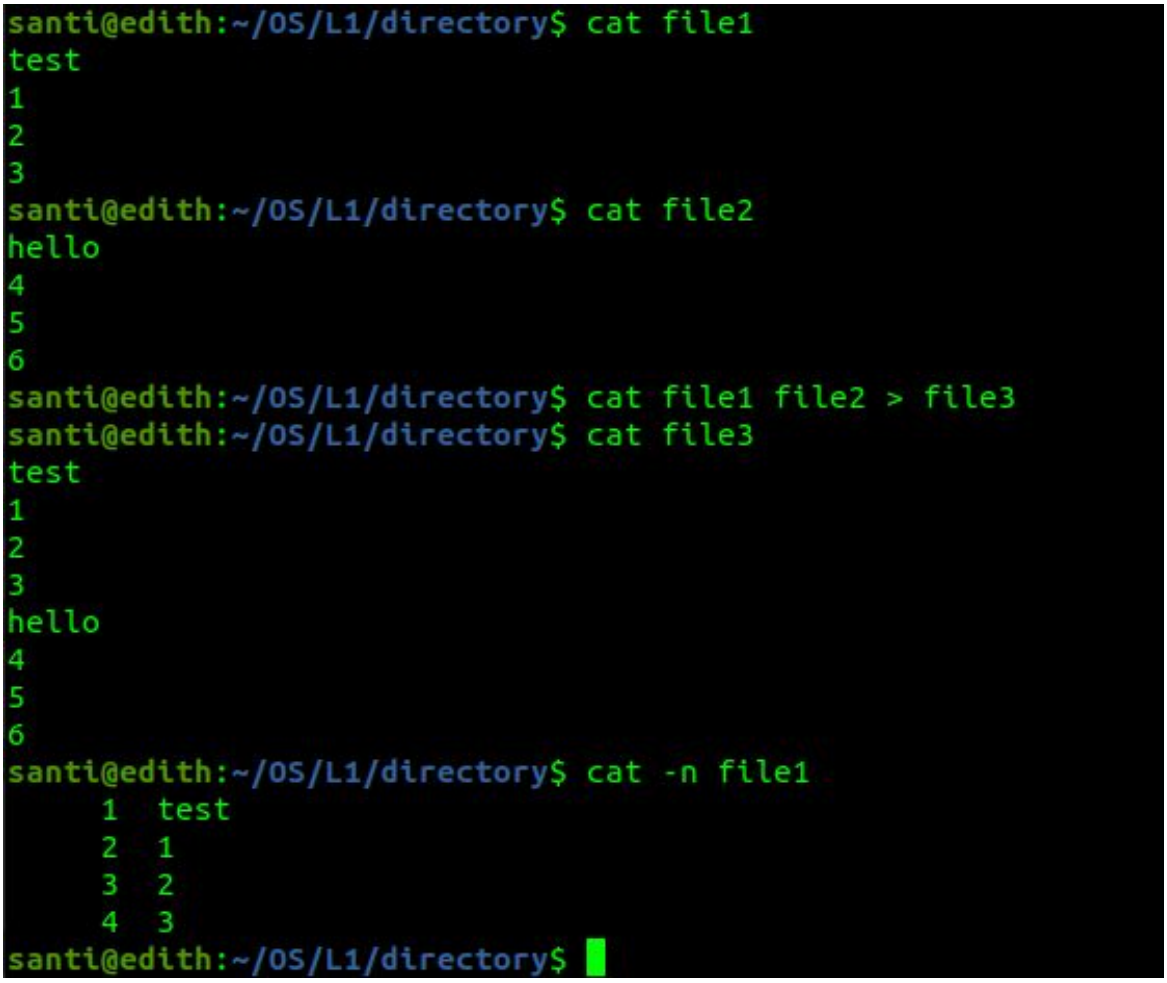
```
santi@edith:~/OS/L1/directory$ ls
file1  file2  new.txt  test.txt
santi@edith:~/OS/L1/directory$ mv -i test.txt new.txt
mv: overwrite 'new.txt'? y
santi@edith:~/OS/L1/directory$ ls
file1  file2  new.txt
santi@edith:~/OS/L1/directory$ mv -f new.txt file2
santi@edith:~/OS/L1/directory$ mv -v file2 test
renamed 'file2' -> 'test'
santi@edith:~/OS/L1/directory$
```

30. cat command

`cat file1 file2` : view multiple files at the same time

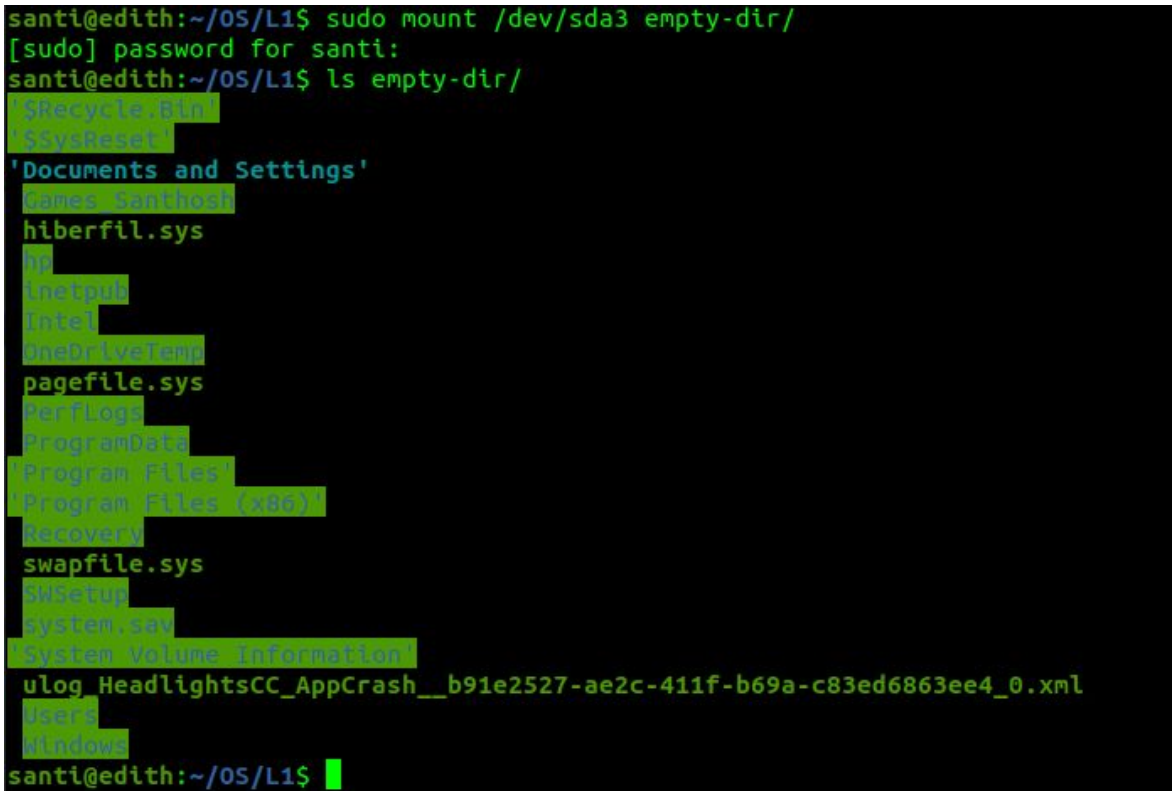
`cat file1 file2 > file3` : tcopy content of file1 followed by content of file2 into file3

`cat -n filename` : view file with line numbers



31. mount command

`mount /dev/sdb1 dirname` : mount sdb1 at the given directory



32. chmod command

`chmod <who><operand><options> files` : sets/changes rwx permissions

- <who> can be <u> for user, <g> for group, <o> for others, <a> for all
- <operand> can be <=> to set the options, <+> to add the options, <-> to remove the options
- <options> can be a combination of r,w,x eg: r, rw, rx, x, empty string, etc

chmod command can also be used this way : `chmod <x><y><z> file` where x,y,z are numbers from 0 to 7. x represents the permission for the user, y the permissions for the group and z for others. x,y,z converted to binary represent the rwx permissions. For eg, x = 6 (110) means the user has r and w permissions. For example, `chmod 777 file` allows rwx permissions to the user, group and others.

```
santi@edith:~/OS/L1$ cat sample1.txt sample2.txt > Input
santi@edith:~/OS/L1$ cp sample2.txt sample.txt
santi@edith:~/OS/L1$ cat sample2.txt >> sample1.txt
santi@edith:~/OS/L1$ ls -l sample.txt
-rw-rw-r-- 1 santi santi 591 Aug 23 15:04 sample.txt
santi@edith:~/OS/L1$ chmod u= sample.txt
santi@edith:~/OS/L1$ ls -l sample.txt
----rw-r-- 1 santi santi 591 Aug 23 15:04 sample.txt
santi@edith:~/OS/L1$
```

33. chown command

`chown user filename` : chown (short for change owner) changes the ownership of the given file to the given user

```
santi@edith:~/OS/L1/directory$ ls -l
total 4
-rw-rw-r-- 1 santi santi 0 Aug 23 13:35 file1
-rw-rw-r-- 1 santi santi 6 Aug 23 13:35 file2
santi@edith:~/OS/L1/directory$ sudo chown root file1
[sudo] password for santi:
santi@edith:~/OS/L1/directory$ ls -l
total 4
-rw-rw-r-- 1 root  santi 0 Aug 23 13:35 file1
-rw-rw-r-- 1 santi santi 6 Aug 23 13:35 file2
santi@edith:~/OS/L1/directory$
```

34. passwd command

`passwd` : change password of current user

`passwd user` : root user can change password of any user with this command

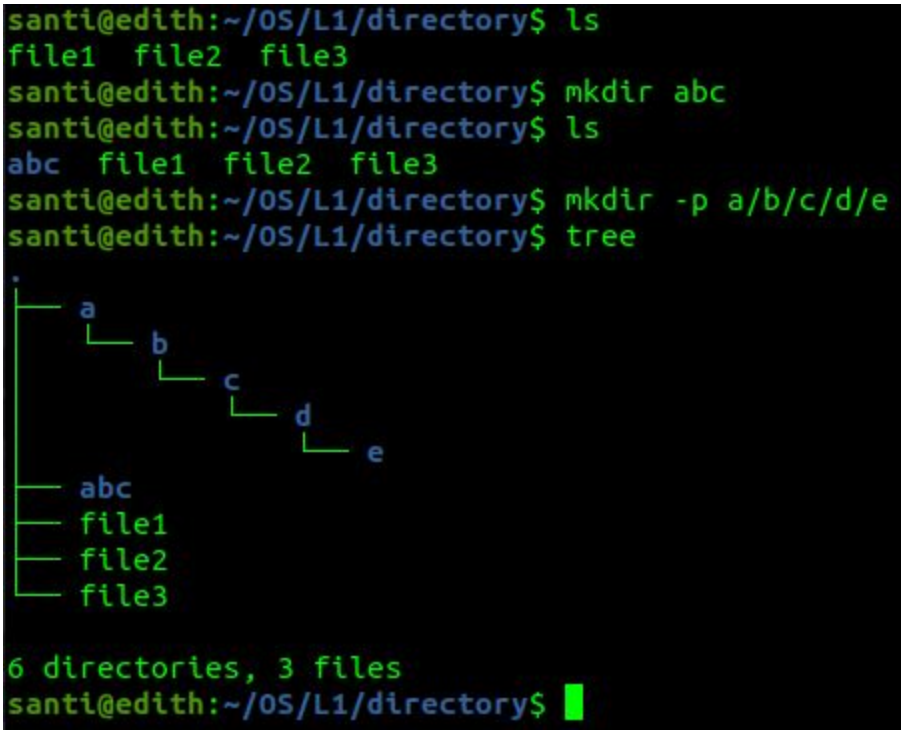
`passwd -d user` : root uset can disable password of any user

```
santi@edith:~/OS/L1$ passwd
Changing password for santi.
Current password:
New password:
Retype new password:
passwd: password updated successfully
santi@edith:~/OS/L1$ sudo passwd -d santi
passwd: password expiry information changed.
santi@edith:~/OS/L1$
```

35. mkdir command

`mkdir dirname` : make a new directory with the specified name

`mkdir -p dir1/dir2/dir3/dir4/` : create nested directories using one mkdir command



36. ifconfig command

`ifconfig -a` : view all the network interfaces and their status

`ifconfig eth0 up` : start eth0 interface

`ifconfig eth0 down` : stop eth0 interface

```
santi@edith:~/OS/L1/directory$ ifconfig -a
eno1: flags=4098<BROADCAST,MULTICAST>  mtu 1500
        ether ec:8e:b5:56:79:7f  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

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 5343  bytes 523913 (523.9 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 5343  bytes 523913 (523.9 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlo1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.43.135  netmask 255.255.255.0  broadcast 192.168.43.255
        inet6 fe80::5c44:1bc0:be2b:f93c  prefixlen 64  scopeid 0x20<link>
        ether 30:e3:7a:0e:30:bc  txqueuelen 1000  (Ethernet)
        RX packets 1141132  bytes 615723569 (615.7 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 420496  bytes 72500092 (72.5 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

santi@edith:~/OS/L1/directory$ ifconfig eno1 up
SIOCSIFFLAGS: Operation not permitted
santi@edith:~/OS/L1/directory$ sudo ifconfig eno1 up
santi@edith:~/OS/L1/directory$ sudo ifconfig eno1 down
santi@edith:~/OS/L1/directory$
```

37. uname command

`uname -a` : displays important information about the system such as — Kernel name, Host name, Kernelrelease number, Processor type, etc.

```
santi@edith:~/OS/L1/directory$ uname -a
Linux edith 5.4.0-42-generic #46-Ubuntu SMP Fri Jul 10 00:24:02 UTC 2020 x86_64 x86_64
x86_64 GNU/Linux
```

38. whereis command

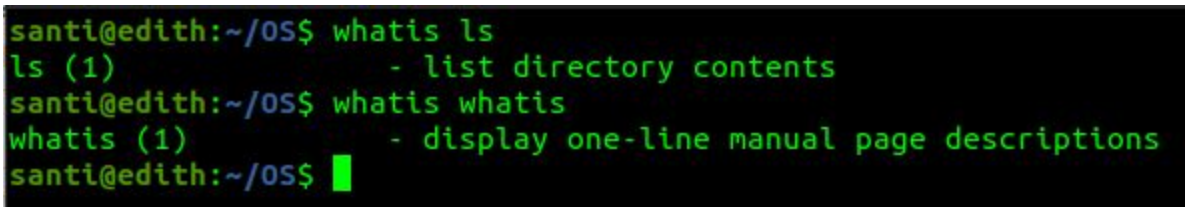
`whereis commandname` : see where the given command exists

`whereis -u -B /directory -f executable` : searches for the given executable in the given directory and displays it if it is available

```
santi@edith:~/OS/L1/directory$ whereis ls
ls: /usr/bin/ls /usr/share/man/man1/ls.1.gz
```

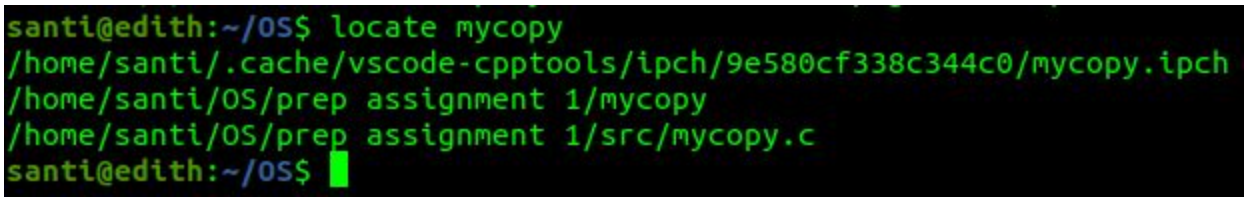
39. whatis command

`whatis commandname` : displays 1 line description of the given command



40. locate command

`locate filename` or `locate <text>` : displays full path of files, directories from the computer which match the given text

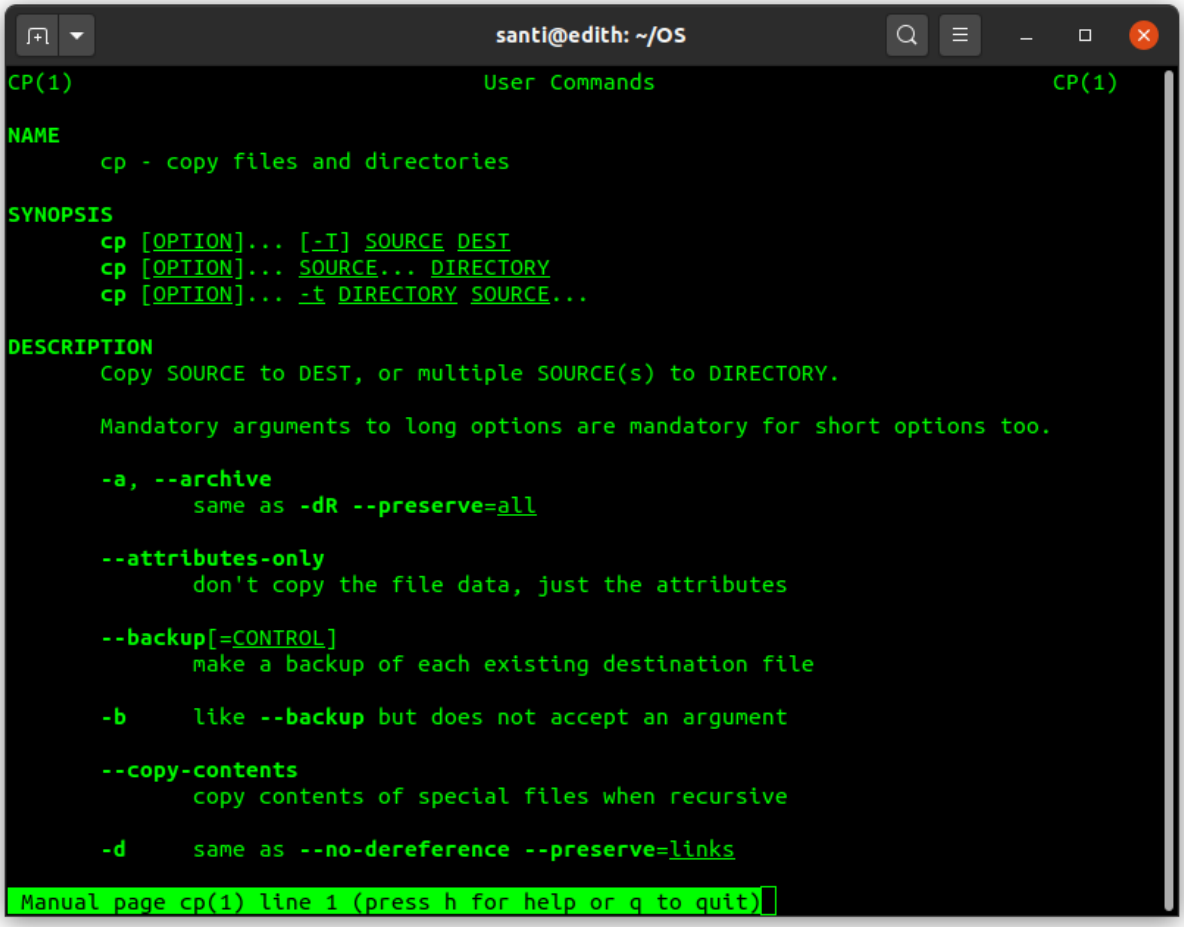


41. man command

`man command` : view manual page for the command

`man <section-number> command` : view specific section of man page where <section-number> is:

1. General commands
2. System calls
3. C library functions
4. Specialfiles (usually devices, those found in /dev) and drivers
5. File formats and conventions
6. Games and screensavers
7. Miscellaneous
8. System administration commands and daemons


santi@edith: ~/OS
CP(1) User Commands CP(1)
NAME
cp - copy files and directories
SYNOPSIS
cp [OPTION]... [-T] SOURCE DEST
cp [OPTION]... SOURCE... DIRECTORY
cp [OPTION]... -t DIRECTORY SOURCE...
DESCRIPTION
Copy SOURCE to DEST, or multiple SOURCE(s) to DIRECTORY.
Mandatory arguments to long options are mandatory for short options too.
-a, --archive
same as -dR --preserve=all
--attributes-only
don't copy the file data, just the attributes
--backup[=CONTROL]
make a backup of each existing destination file
-b like --backup but does not accept an argument
--copy-contents
copy contents of special files when recursive
-d same as --no-dereference --preserve=links
Manual page cp(1) line 1 (press h for help or q to quit)

42. tail command

`tail filename` : print the last 10 lines of a file by default

`tail -n N filename` : print N number of lines from the file

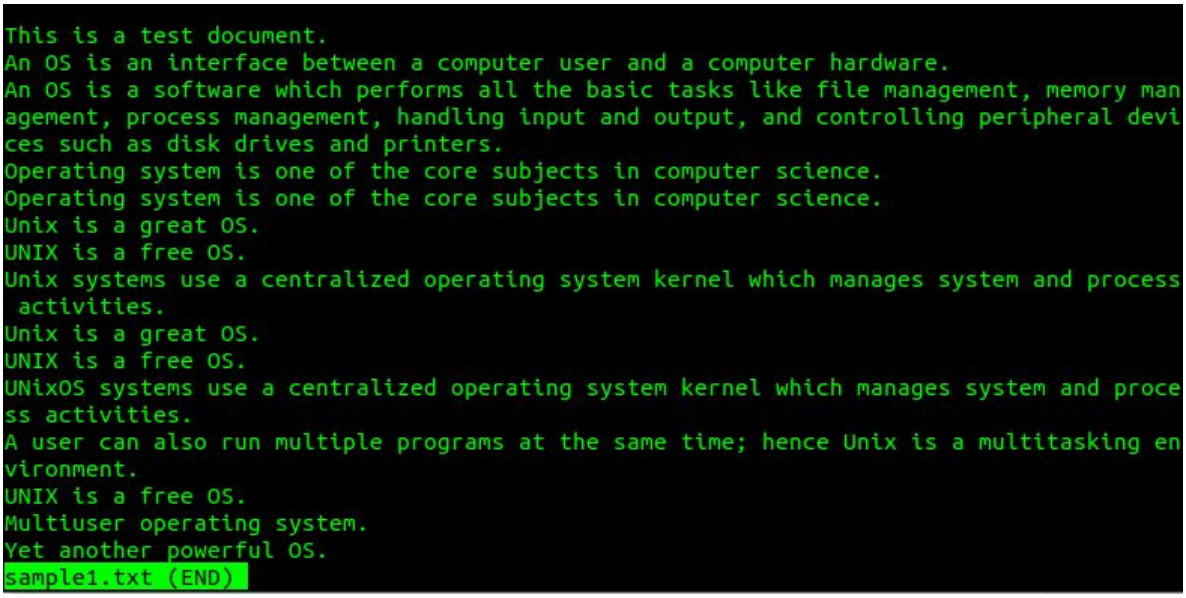
`tail -f filename` : view the file in realtime

```
santi@edith:~/OS/L1$ tail Input.txt
Multiuser operating system.
Yet another powerful OS.
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis
 Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and
 BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multius
er system.
A user can also run multiple programs at the same time; hence Unix is a multitasking en
vironment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.

santi@edith:~/OS/L1$ tail -n 2 Input.txt
Yet another powerful OS.

santi@edith:~/OS/L1$ tail -f Input.txt
Multiuser operating system.
Yet another powerful OS.
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis
 Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and
 BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multius
er system.
A user can also run multiple programs at the same time; hence Unix is a multitasking en
vironment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
```

43. less command

`less filename` : view contents of file

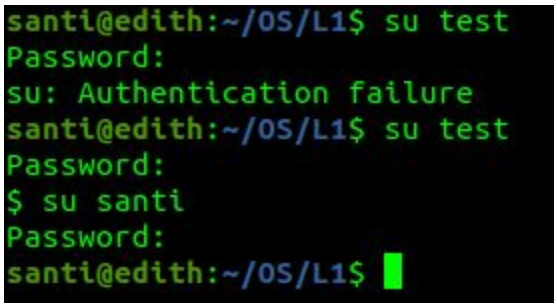
Use ^f or ^b to move 1 window forward or backward while viewing



44. su command

`su username` : login to a specified user account

`su -s 'shellname' username` : login to a specified user account, and execute the specified shell instead of the default shell



45. mysql command

`mysql -u root -p -h 192.168.1.2` : connect to a remote mysql db

`mysql -u root -p` : connect to a local mysql db

46. yum command

`yum install httpd` : install apache using rpm

`yum update httpd` : upgrade apache using rpm

`yum remove httpd` : uninstall/remove apache using rpm

47. rpm command

`rpm -ivh httpd-2.2.3-22.0.1.el5.i386.rpm` : install apache using rpm

`rpm -uvh httpd-2.2.3-22.0.1.el5.i386.rpm` : upgrade apache using rpm

`rpm -ev httpd` : uninstall/remove apache using rpm

48. ping command

`ping google.com` : ping a remote host

`ping -c 3 google.com` : ping a remote host bby sending exactly 3 packets

```
santi@edith:~/OS/L1$ ping google.com
PING google.com (142.250.67.46) 56(84) bytes of data.
64 bytes from maa05s12-in-f14.1e100.net (142.250.67.46): icmp_seq=1 ttl=118 time=6.81 ms
64 bytes from maa05s12-in-f14.1e100.net (142.250.67.46): icmp_seq=2 ttl=118 time=7.69 ms
64 bytes from maa05s12-in-f14.1e100.net (142.250.67.46): icmp_seq=3 ttl=118 time=8.00 ms
64 bytes from maa05s12-in-f14.1e100.net (142.250.67.46): icmp_seq=4 ttl=118 time=7.74 ms
^C
--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 6.813/7.560/7.998/0.446 ms
santi@edith:~/OS/L1$ ping -c 3 google.com
PING google.com (142.250.67.46) 56(84) bytes of data.
64 bytes from maa05s12-in-f14.1e100.net (142.250.67.46): icmp_seq=1 ttl=118 time=7.12 ms
64 bytes from maa05s12-in-f14.1e100.net (142.250.67.46): icmp_seq=2 ttl=118 time=8.16 ms
64 bytes from maa05s12-in-f14.1e100.net (142.250.67.46): icmp_seq=3 ttl=118 time=8.09 ms

--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 7.117/7.790/8.160/0.476 ms
santi@edith:~/OS/L1$
```

49. date command

`date` : display system date,time

`date -s "01/31/2010 23:59:53"` : set system date and time

```
santi@edith:~/OS/L1$ date
Friday 28 August 2020 09:43:04 PM IST
santi@edith:~/OS/L1$
```

50. wget command

`wget url` : download the file in the given url

`wget -o filename url` : download the file and save it with a different filename

```
santi@edith:~/OS/L1/directory$ ls
file1  file2  file3
santi@edith:~/OS/L1/directory$ wget https://speed.hetzner.de/100MB.bin
--2020-08-28 22:29:50--  https://speed.hetzner.de/100MB.bin
Resolving speed.hetzner.de (speed.hetzner.de)... 88.198.248.254, 2a01:4f8:0:59ed::2
Connecting to speed.hetzner.de (speed.hetzner.de)|88.198.248.254|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 104857600 (100M) [application/octet-stream]
Saving to: '100MB.bin'

100MB.bin             100%[=========================>] 100.00M  1.16MB/s    in 87s

2020-08-28 22:31:19 (1.15 MB/s) - '100MB.bin' saved [104857600/104857600]

santi@edith:~/OS/L1/directory$ ls
100MB.bin  file1  file2  file3
santi@edith:~/OS/L1/directory$
```

Questions 2 to 41

2. Create a directory and create a file inside that directory.

`mkdir directory` : create a directory named directory

`cd directory` : change current directory to directory

`touch file1` : create an empty file named file1

`echo "hello" > file2` : create a file named file2 and write "hello" in the file

```
santi@edith:~/OS/L1$ mkdir directory
santi@edith:~/OS/L1$ cd directory
santi@edith:~/OS/L1/directory$ touch file1
santi@edith:~/OS/L1/directory$ echo "hello" > file2
santi@edith:~/OS/L1/directory$ ls
file1  file2
santi@edith:~/OS/L1/directory$ cat file1
santi@edith:~/OS/L1/directory$ cat file2
hello
santi@edith:~/OS/L1/directory$
```

3. List the files and directories that are empty in a working directory.

`find text` : search for files/directories in the current directory that match the text
`find -empty` : -empty is an option to print empty directories or files
`find -type f` or `find -type d` : -type is an option to print directories or folders alone. -type d prints directories, -type f prints files.
`find -empty -type f` or `find -empty -type d` : we can use the combination of these options as given to print empty directories or empty folders separately

```
santi@edith:~/OS/L1$ find *.pdf
50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf
Exercise-I LINUX COMMANDS.pdf
linuxcommands.pdf
santi@edith:~/OS/L1$ find -empty
./directory/file1
./empty-dir
santi@edith:~/OS/L1$ find -empty -type d
./empty-dir
santi@edith:~/OS/L1$ find -empty -type f
./directory/file1
santi@edith:~/OS/L1$
```

4. Show commands to delete empty and non-empty directory.

`rmdir dirname` or `rm -d dirname` : deletes the directory named dirname only if it is empty
`rm -r dirname` : deletes the directory named dirname and all the subdirectories and files inside it

```
santi@edith:~/OS/L1$ rmdir empty-dir
santi@edith:~/OS/L1$ rmdir test
rmdir: failed to remove 'test': Directory not empty
santi@edith:~/OS/L1$ rm -d test
rm: cannot remove 'test': Directory not empty
santi@edith:~/OS/L1$ rm -r test
santi@edith:~/OS/L1$
```

5. Find the location of the input files using locate and find command.

`locate text` : displays full path of files, directories **from the entire computer** which match the given text

`find text` : displays full path of files, directories **from the current directory** which match the given text

```
santi@edith:~/OS$ locate L1/
/home/santi/OS/L1/50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf
/home/santi/OS/L1/Exercise-I LINUX COMMANDS.pdf
/home/santi/OS/L1/directory
/home/santi/OS/L1/directory/file1
/home/santi/OS/L1/directory/file2
/home/santi/OS/L1/empty-dir
/home/santi/OS/L1/linuxcommands.pdf
/home/santi/OS/L1/sample1.txt
/home/santi/OS/L1/sample2.txt
santi@edith:~/OS$ find sample*
find: ‘sample*’: No such file or directory
santi@edith:~/OS$ find ./*/sample*
./L1/sample1.txt
./L1/sample2.txt
santi@edith:~/OS$ find L1/*.pdf
L1/50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf
L1/Exercise-I LINUX COMMANDS.pdf
L1/linuxcommands.pdf
santi@edith:~/OS$
```

6. View the user permissions and ownership of the files in the current directory and change the ownership of some selected files to another user.

`ls -l` : ls (short for list) is a command used to display the contents of the current directory. -l option displays the content in a long listing format which contains user permissions and ownership of the files.

`chown user filename` : chown (short for change owner) changes the ownership of the given file to the given user

```
santi@edith:~/OS/L1/directory$ ls -l
total 4
-rw-rw-r-- 1 santi santi 0 Aug 23 13:35 file1
-rw-rw-r-- 1 santi santi 6 Aug 23 13:35 file2
santi@edith:~/OS/L1/directory$ sudo chown root file1
[sudo] password for santi:
santi@edith:~/OS/L1/directory$ ls -l
total 4
-rw-rw-r-- 1 root  santi 0 Aug 23 13:35 file1
-rw-rw-r-- 1 santi santi 6 Aug 23 13:35 file2
santi@edith:~/OS/L1/directory$
```

7. List all the files in the current directory and subdirectories.

`ls -R` : -R (stands for recursive) option in ls displays the files in current directory and the subdirectories.

```
santi@edith:~/OS/L1$ ls -R
.:
'50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf'
 directory
 empty-dir
'Exercise-I LINUX COMMANDS.pdf'
 linuxcommands.pdf
 sample1.txt
 sample2.txt

./directory:
file1  file2

./empty-dir:
santi@edith:~/OS/L1$
```

8. Concatenate the two input files: “sample1.txt” and “sample2.txt” and save it to a new file named “Input”.

`cat file1 file2 > file3` : this command copies content of file1 followed by content of file2 into file3

```
santi@edith:~/OS/L1$ cat sample1.txt sample2.txt > Input
```

9. Copy the contents of file ‘sample2.txt’ to ‘sample.txt’

`cp source dest` : makes a copy of the file ‘source’ in a new file name ‘dest’

```
santi@edith:~/OS/L1$ cp sample2.txt sample.txt
```

10. Append the file contents of input file ‘sample2.txt’ to the end of the first input file ‘sample1.txt’.

`cat file1 >> file2` : appends the contents of file1 at the end of file2

```
santi@edith:~/OS/L1$ cat sample2.txt >> sample1.txt
```

11. Remove the permission for the users to read, write and execute the file ‘sample.txt’

`chmod <who><operand><options> files` : sets/changes rwx permissions

- <who> can be <u> for user, <g> for group, <o> for others, <a> for all
- <operand> can be <=> to set the options, <+> to add the options, <-> to remove the options
- <options> can be a combination of r,w,x eg: r, rw, rx, x, empty string, etc

So to remove the permission for the users to read, write and execute the file ‘sample.txt’, the command will be `chmod u= sample.txt`

chmod command can also be used this way : `chmod <x><y><z> file` where x,y,z are numbers from 0 to 7. x represents the permission for the user, y the permissions for the group and z for others. x,y,z converted to binary represent the rwx permissions. For eg, x = 6 (110) means the user has r and w permissions. For example, `chmod 777 file` allows rwx permissions to the user, group and others.

```
santi@edith:~/OS/L1$ ls -l sample.txt
-rw-rw-r-- 1 santi santi 591 Aug 23 15:04 sample.txt
santi@edith:~/OS/L1$ chmod u= sample.txt
santi@edith:~/OS/L1$ ls -l sample.txt
----rw-r-- 1 santi santi 591 Aug 23 15:04 sample.txt
santi@edith:~/OS/L1$
```

12. Display the current date with the day of week, month, time and the year.

`date` : displays day of the week, date with month and year, time

```
santi@edith:~$ date
Sunday 23 August 2020 03:11:36 PM IST
```

13. Show the calendar of previous, current and next month.

`cal` : displays the calendar of the current month
`cal -Ax` : displays the calendar of the current month and x months after it
`cal -Bx` : displays the calendar of the current month and x months before it
`cal -3` : displays the calendar of the previous, current and next months
`cal -A1 -B1` can also be used for the same purpose

```
santi@edith:~$ cal -3
                                2020
        July                 August               September
Su Mo Tu We Th Fr Sa  Su Mo Tu We Th Fr Sa  Su Mo Tu We Th Fr Sa
          1  2  3  4                     1         1  2  3  4  5
 5  6  7  8  9 10 11   2  3  4  5  6  7  8   6  7  8  9 10 11 12
12 13 14 15 16 17 18   9 10 11 12 13 14 15  13 14 15 16 17 18 19
19 20 21 22 23 24 25  16 17 18 19 20 21 22  20 21 22 23 24 25 26
26 27 28 29 30 31     23 24 25 26 27 28 29  27 28 29 30
                      30 31
santi@edith:~$
```

14. Sort the contents of the file ‘sample1.txt’ in alphabetical order

`sort filename` : sorts and displays the contents of the given file

```
santi@edith:~/OS/L1$ sort sample1.txt
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, mem
ory management, process management, handling input and output, and controlling p
eripheral devices such as disk drives and printers.
A user can also run multiple programs at the same time; hence Unix is a multitas
king environment.
Multiuser operating system.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
This is a test document.
UNIX is a free OS.
UNIX is a free OS.
UNIX is a free OS.
Unix is a great OS.
Unix is a great OS.
UNixOS systems use a centralized operating system kernel which manages system an
d process activities.
Unix systems use a centralized operating system kernel which manages system and
process activities.
Yet another powerful OS.
santi@edith:~/OS/L1$
```

15. Erase duplicate records in the file ‘sample1.txt’ and display only the unique records

`sort -u sample1.txt` : sorts and displays only the unique records in the file

`sort -u sample1.txt | cat > sample1.txt` can be used to replace the contents of sample1.txt with the uniquely sorted records.

```
santi@edith:~/OS/L1$ sort -u sample1.txt
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, mem
ory management, process management, handling input and output, and controlling p
eripheral devices such as disk drives and printers.
A user can also run multiple programs at the same time; hence Unix is a multitas
king environment.
Multiuser operating system.
Operating system is one of the core subjects in computer science.
This is a test document.
UNIX is a free OS.
Unix is a great OS.
UNixOS systems use a centralized operating system kernel which manages system an
d process activities.
Unix systems use a centralized operating system kernel which manages system and
process activities.
Yet another powerful OS.
santi@edith:~/OS/L1$
```

16. Add line numbers to the file ‘sample2.txt’

`nl filename` : adds line number and prints the file content, doesn’t add line number for blank lines

`nl -b a filename` or `cat -n filename` : adds line number and prints the file content, adds line number for blank lines

`nl -b a file1 | cat > file2` or `cat -n filename | cat > file2` : add line number in the contents of file1 and save in file2. So, for the given question,

`cat -n sample2.txt | cat > sample2.txt` can be used

```
santi@edith:~/OS/L1$ nl test.txt
     1  line1

     2  line2 after blankline
     3  line3
santi@edith:~/OS/L1$ nl -b a test.txt
     1  line1
     2
     3  line2 after blankline
     4  line3
santi@edith:~/OS/L1$ cat -n test.txt
     1  line1
     2
     3  line2 after blankline
     4  line3
```

17. Find out whether the two pairs of input files are identical or not.

`cmp file1 file2` : compare the two files

```
santi@edith:~/OS/L1$ cmp sample1.txt sample2.txt
sample1.txt sample2.txt differ: byte 1, line 1
santi@edith:~/OS/L1$ cmp sample2.txt sample.txt
santi@edith:~/OS/L1$
```

18. Show how the input file “sample1.txt” differs line by line from “sample2.txt” in context and unified mode.

`diff -u file1 file2` : compare the two files and display the differences in unified mode

`diff -c file1 file2` : compare the two files and display the differences in context mode

```
santi@edith:~/OS/L1$ diff -u sample1.txt sample2.txt
--- sample1.txt 2020-08-23 15:30:11.982470381 +0530
+++ sample2.txt 2020-08-23 15:16:39.108951312 +0530
@@ -1,15 +1,8 @@
-This is a test document.
-An OS is an interface between a computer user and a computer hardware.
-An OS is a software which performs all the basic tasks like file management, memory management, proc
ess management, handling input and output, and controlling peripheral devices such as disk drives and
 printers.
-Operating system is one of the core subjects in computer science.
-Operating system is one of the core subjects in computer science.
-Unix is a great OS.
-UNIX is a free OS.
-Unix systems use a centralized operating system kernel which manages system and process activities.
-Unix is a great OS.
-UNIX is a free OS.
-UNixOS systems use a centralized operating system kernel which manages system and process activities
.
-A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
-UNIX is a free OS.
-Multiuser operating system.
-Yet another powerful OS.
+Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Dou
glas McIlroy, and Joe Ossanna at Bell Labs.
+There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a fe
w examples. Linux is also a flavor of Unix which is freely available.
+Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
+A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
+UNIX is a free OS.
+Multiuser operating system.
+Yet another powerful OS.
+
```

```
santi@edith:~/OS/L1$ diff -c sample1.txt sample2.txt
*** sample1.txt 2020-08-23 15:30:11.982470381 +0530
--- sample2.txt 2020-08-23 15:16:39.108951312 +0530
***************
*** 1,15 ****
! This is a test document.
! An OS is an interface between a computer user and a computer hardware.
! An OS is a software which performs all the basic tasks like file management, memory management, pro
cess management, handling input and output, and controlling peripheral devices such as disk drives an
d printers.
! Operating system is one of the core subjects in computer science.
! Operating system is one of the core subjects in computer science.
! Unix is a great OS.
! UNIX is a free OS.
! Unix systems use a centralized operating system kernel which manages system and process activities.
! Unix is a great OS.
! UNIX is a free OS.
! UNixOS systems use a centralized operating system kernel which manages system and process activitie
s.
! A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
! UNIX is a free OS.
! Multiuser operating system.
! Yet another powerful OS.
--- 1,8 ----
! Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Do
uglas McIlroy, and Joe Ossanna at Bell Labs.
! There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a f
ew examples. Linux is also a flavor of Unix which is freely available.
! Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
! A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
! UNIX is a free OS.
! Multiuser operating system.
! Yet another powerful OS.
!
santi@edith:~/OS/L1$
```

19. Solve the arithmetic expression: ((8+12)*(5-3))/2 using linux commands

```
santi@edith:~/OS/L1$ echo "((8+12)*(5-3))/2" | bc
20
santi@edith:~/OS/L1$ x=$((((8+12)*(5-3))/2))
santi@edith:~/OS/L1$ echo x
x
santi@edith:~/OS/L1$ echo $x
20
santi@edith:~/OS/L1$
```

20. Cut and display the first 10 characters of every line of the file “Input.txt”.

```
santi@edith:~/OS/L1$ cut -c 1-10 Input.txt
This is a
An OS is a
An OS is a
Operating
Operating
Unix is a
UNIX is a
Unix syste
Unix is a
UNIX is a
UNixOS sys
A user can
UNIX is a
Multiuser
Yet anothe
Unix was o
There are
Several pe
A user can
UNIX is a
Multiuser
Yet anothe
```

21. Print the name of the current working directory.

```
santi@edith:~/OS/L1$ pwd
/home/santi/OS/L1
```

22. Process Status

a. List all the running processes with their corresponding PIDs.

b. List the processes that are not associated with the terminal.

c. List the processes that are associated with the terminal.

`ps -a` : for a. `ps -aN` : for b. `ps -T` : for c.

```
santi@edith:~/OS/L1$ ps -a
    PID TTY          TIME CMD
   1052 tty1     00:00:05 Xorg
   1291 tty1     00:00:00 gnome-session-b
   1652 tty2     00:03:45 Xorg
   1763 tty2     00:00:00 gnome-session-b
   4801 pts/0    00:00:00 ps
santi@edith:~/OS/L1$ ps -aN | head -5
    PID TTY          TIME CMD
      1 ?        00:00:10 systemd
      2 ?        00:00:00 kthreadd
      3 ?        00:00:00 rcu_gp
      4 ?        00:00:00 rcu_par_gp
santi@edith:~/OS/L1$ ps -T
    PID    SPID TTY          TIME CMD
   4284    4284 pts/0    00:00:00 bash
   4805    4805 pts/0    00:00:00 ps
santi@edith:~/OS/L1$
```

23. Print the number of characters, number of lines and number of words of all the given input files.

```
santi@edith:~/OS/L1$ wc --chars --lines --words sample1.txt sample2.txt Input.txt
  15  149  893 sample1.txt
   8  102  591 sample2.txt
  23  251 1484 Input.txt
  46  502 2968 total
santi@edith:~/OS/L1$
```

24. Print the length of the longest line from all the input files.

```
santi@edith:~/OS/L1$ wc -L sample1.txt sample2.txt Input.txt
 211 sample1.txt
 169 sample2.txt
 211 Input.txt
 211 total
santi@edith:~/OS/L1$
```

25. Move the contents of the input file sample.txt to a new file.

```
santi@edith:~/OS/L1$ cat sample.txt > newfile.txt
santi@edith:~/OS/L1$ cat newfile.txt
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ri
tchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BS
D are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser
system.
A user can also run multiple programs at the same time; hence Unix is a multitasking envir
onment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
```

26. Copy the contents of one directory to another directory

```
santi@edith:~/OS/L1$ ls directory/
file1  file2
santi@edith:~/OS/L1$ cp -r directory dircopy
santi@edith:~/OS/L1$ ls dircopy/
file1  file2
santi@edith:~/OS/L1$
```

27. Reverse the lines of the two input files and concatenate the file contents using a single command.

`cat file` : displays the contents of file

`tac file` : displays the contents of file in reverse order

`tac file1 file2 > file3` : reverse the lines of file1, file2 concatenate the file contents and save it in file3

```
santi@edith:~/OS/L1/directory$ cat file1
test
1
2
3
santi@edith:~/OS/L1/directory$ tac file1
3
2
1
test
santi@edith:~/OS/L1/directory$ tac file1 file2 > file4
santi@edith:~/OS/L1/directory$ cat file4
3
2
1
test
6
5
4
hello
santi@edith:~/OS/L1/directory$
```

28. Delete all the files with *.txt extension from the working directory using yes command

```
santi@edith:~/OS/L1/directory$ ls
file1  file2  file3  file4  test2.txt  test3.txt  test4.txt  test.txt
santi@edith:~/OS/L1/directory$ yes | rm *.txt
santi@edith:~/OS/L1/directory$ ls
file1  file2  file3  file4
santi@edith:~/OS/L1/directory$
```

29. Given the input file “sample1.txt”, print the number of the lines that match the pattern “system”.

```
santi@edith:~/OS/L1$ grep system sample1.txt | wc -l
5
```

30. Having sample1 file as input, print the matched lines that contain the pattern “Unix” as whole words.

```
santi@edith:~/OS/L1$ grep -w Unix sample1.txt
Unix is a great OS.
Unix systems use a centralized operating system kernel which manages system and proc
ess activities.
Unix is a great OS.
A user can also run multiple programs at the same time; hence Unix is a multitasking
 environment.
```

31. Print the lines from “sample1.txt” that do not match the pattern “OS”.

```
santi@edith:~/OS/L1$ grep -v OS sample1.txt
This is a test document.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix systems use a centralized operating system kernel which manages system and proc
ess activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking
 environment.
Multiuser operating system.
santi@edith:~/OS/L1$
```

32. Fetch the files that contain the word “OS”, “Operating System”, “Operating Systems” with its respective line number. (Ignore the case).

```
grep -e OS -e “Operating system” -e “Operating Systems” -i -R -n
```

```
santi@edith:~/OS/L1$ grep -e OS -e "Operating system" -e "Operating Systems" -l -R -n
Binary file 50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf matches
sample.txt:1:Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell
 Labs.
sample.txt:5:UNIX is a free OS.
sample.txt:6:Multiuser operating system.
sample.txt:7:Yet another powerful OS.
Binary file linuxcommands.pdf matches
Binary file Exercise-I LINUX COMMANDS.pdf matches
sample2.txt:1:Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bel
l Labs.
sample2.txt:5:UNIX is a free OS.
sample2.txt:6:Multiuser operating system.
sample2.txt:7:Yet another powerful OS.
Input.txt:2:An OS is an interface between a computer user and a computer hardware.
Input.txt:3:An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and out
put, and controlling peripheral devices such as disk drives and printers.
Input.txt:4:Operating system is one of the core subjects in computer science.
Input.txt:5:Operating system is one of the core subjects in computer science.
Input.txt:6:Unix is a great OS.
Input.txt:7:UNIX is a free OS.
Input.txt:8:Unix systems use a centralized operating system kernel which manages system and process activities.
Input.txt:9:Unix is a great OS.
Input.txt:10:UNIX is a free OS.
Input.txt:11:UNixOS systems use a centralized operating system kernel which manages system and process activities.
Input.txt:13:UNIX is a free OS.
Input.txt:14:Multiuser operating system.
Input.txt:15:Yet another powerful OS.
Input.txt:16:Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell
 Labs.
Input.txt:20:UNIX is a free OS.
Input.txt:21:Multiuser operating system.
Input.txt:22:Yet another powerful OS.
sample1.txt:2:An OS is an interface between a computer user and a computer hardware.
sample1.txt:3:An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and o
utput, and controlling peripheral devices such as disk drives and printers.
sample1.txt:4:Operating system is one of the core subjects in computer science.
sample1.txt:5:Operating system is one of the core subjects in computer science.
sample1.txt:6:Unix is a great OS.
sample1.txt:7:UNIX is a free OS.
sample1.txt:8:Unix systems use a centralized operating system kernel which manages system and process activities.
sample1.txt:9:Unix is a great OS.
sample1.txt:10:UNIX is a free OS.
```

33. Having “sample1.txt” and “core” as the input and pattern respectively, along with the matched line print three lines before and after the pattern match.

```
santi@edith:~/OS/L1$ grep core -B3 sample1.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory
management, process management, handling input and output, and controlling peripher
l devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
santi@edith:~/OS/L1$ grep core -A3 sample1.txt
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and proc
ess activities.
santi@edith:~/OS/L1$
```

34. Find and replace the string “OS” with “Operating System”.

```
santi@edith:~/OS/L1$ cat sample1.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, mem
ory management, process management, handling input and output, and controlling p
eripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and
process activities.
Unix is a great OS.
UNIX is a free OS.
UNixOS systems use a centralized operating system kernel which manages system an
d process activities.
A user can also run multiple programs at the same time; hence Unix is a multitas
king environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
santi@edith:~/OS/L1$
```

```
santi@edith:~/OS/L1$ sed 's/OS/Operating System/' sample1.txt
This is a test document.
An Operating System is an interface between a computer user and a computer hardw
are.
An Operating System is a software which performs all the basic tasks like file m
anagement, memory management, process management, handling input and output, and
 controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great Operating System.
UNIX is a free Operating System.
Unix systems use a centralized operating system kernel which manages system and
process activities.
Unix is a great Operating System.
UNIX is a free Operating System.
UNixOperating System systems use a centralized operating system kernel which man
ages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitas
king environment.
UNIX is a free Operating System.
Multiuser operating system.
Yet another powerful Operating System.
santi@edith:~/OS/L1$
```

35. List only the text files in the current working directory with its corresponding disk space occupied.

```
santi@edith:~/OS/L1$ ls -sh *.txt
4.0K Input.txt    4.0K sample1.txt  4.0K sample.txt
4.0K newfile.txt  4.0K sample2.txt  4.0K test.txt
santi@edith:~/OS/L1$
```

36. Show the last modification time of all the input text files.

```
santi@edith:~/OS/L1$ stat -c '%n: %y' *.txt
Input.txt: 2020-08-27 20:07:15.067884696 +0530
newfile.txt: 2020-08-27 20:30:41.847607981 +0530
sample1.txt: 2020-08-23 15:30:11.982470381 +0530
sample2.txt: 2020-08-23 15:16:39.108951312 +0530
sample.txt: 2020-08-27 19:54:39.235055145 +0530
test.txt: 2020-08-27 20:34:11.908288245 +0530
santi@edith:~/OS/L1$
```

37. Delete the line that has the word “Powerful” from text file “sample2.txt”.

```
santi@edith:~/OS/L1$ cat sample2.txt
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis
Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and
BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuse
r system.
A user can also run multiple programs at the same time; hence Unix is a multitasking env
ironment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
santi@edith:~/OS/L1$ sed '/powerful/d' sample2.txt
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis
Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and
BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuse
r system.
A user can also run multiple programs at the same time; hence Unix is a multitasking env
ironment.
UNIX is a free OS.
Multiuser operating system.
santi@edith:~/OS/L1$
```

38. Print the roll numbers that end with even numbers in the format (COE18B002) up to COE18B050.

```
santi@edith:~$ for ((i=2; i<=9; i++)); do echo "COE18B00$i"; done;
for ((j=10; j<=50; j++)); do echo "COE18B0$j"; done;
COE18B002
COE18B003
COE18B004
COE18B005
COE18B006
COE18B007
COE18B008
COE18B009
COE18B010
COE18B011
COE18B012
COE18B013
COE18B014
COE18B015
COE18B016
COE18B017
COE18B018
COE18B019
COE18B020
```

39. Use filter commands like head, tail, more to view the file contents page by page.

`head file` : displays the first 10 lines of the file

`tail file` : displays the last 10 lines of the file

`head -n <N> file` : displays the first N lines of the file

`tail -n <N> file` : displays the last N lines of the file

`more -<N> file` : displays the contents of the file N lines at a time, press space bar to view more and q to exit

```
santi@edith:~/OS/L1$ head -n 3 sample1.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory mana
gement, process management, handling input and output, and controlling peripheral device
s such as disk drives and printers.
santi@edith:~/OS/L1$ tail -n 3 sample1.txt
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
santi@edith:~/OS/L1$ more -3 sample1.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory mana
gement, process management, handling input and output, and controlling peripheral device
s such as disk drives and printers.
Operating system is one of the core subjects in computer science.
--More--(41%)
```

40. Compress the current working directory contents to a tar file and extract those files from the compressed tar file.

```
santi@edith:~/OS/L1/directory$ tar -zcvf test.tar.gz .
./
./file2
./file1
./file4
./file3
tar: .: file changed as we read it
santi@edith:~/OS/L1/directory$ tar -tvf test.tar.gz
drwxrwxr-x santi/santi       0 2020-08-29 13:05 ./
-rw-rw-r-- santi/santi      12 2020-08-28 21:07 ./file2
-rw-rw-r-- santi/santi      11 2020-08-28 21:06 ./file1
-rw-rw-r-- santi/santi      23 2020-08-29 12:59 ./file4
-rw-rw-r-- santi/santi      23 2020-08-28 21:07 ./file3
santi@edith:~/OS/L1/directory$ tar -xvf test.tar.gz
./
./file2
./file1
./file4
./file3
santi@edith:~/OS/L1/directory$
```

41. Compress the files using zip command.

a. Zip the input file “sample1.txt” as samplezip.zip and remove the file from the current directory after zipping.

```
santi@edith:~/OS/L1$ ls
'50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf'   Input.txt
 directory                                                         linuxcommands.pdf
 directory.zip                                                     sample1.txt
 empty-dir                                                         sample2.txt
'Exercise-I LINUX COMMANDS.pdf'                                    sample.txt
santi@edith:~/OS/L1$ zip -m samplezip.zip sample1.txt
  adding: sample1.txt (deflated 58%)
santi@edith:~/OS/L1$ ls
'50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf'   Input.txt
 directory                                                         linuxcommands.pdf
 directory.zip                                                     sample2.txt
 empty-dir                                                         sample.txt
'Exercise-I LINUX COMMANDS.pdf'                                    samplezip.zip
santi@edith:~/OS/L1$
```

b. Add “sample2.txt” and update the zip archive.

```
santi@edith:~/OS/L1$ unzip -l samplezip.zip
Archive:  samplezip.zip
  Length      Date    Time    Name
---------  ---------- -----   ----
      893  2020-08-23 15:30   sample1.txt
---------                     -------
      893                     1 file
santi@edith:~/OS/L1$ zip -u samplezip.zip sample2.txt
  adding: sample2.txt (deflated 40%)
santi@edith:~/OS/L1$ unzip -l samplezip.zip
Archive:  samplezip.zip
  Length      Date    Time    Name
---------  ---------- -----   ----
      893  2020-08-23 15:30   sample1.txt
      589  2020-08-29 13:14   sample2.txt
---------                     -------
     1482                     2 files
santi@edith:~/OS/L1$
```

c. Zip a directory with all its contents.

```
santi@edith:~/OS/L1$ ls directory
file1  file2  file3  file4
santi@edith:~/OS/L1$ zip -r dir.zip directory
  adding: directory/ (stored 0%)
  adding: directory/file2 (deflated 40%)
  adding: directory/file1 (deflated 58%)
  adding: directory/file4 (deflated 59%)
  adding: directory/file3 (deflated 40%)
santi@edith:~/OS/L1$
```

d. Remove a file from the zip archive

```
santi@edith:~/OS/L1$ unzip -l dir.zip
Archive:  dir.zip
  Length      Date    Time    Name
---------  ---------- -----   ----
        0  2020-08-29 13:42   directory/
      589  2020-08-29 13:50   directory/file2
      893  2020-08-29 13:50   directory/file1
     1484  2020-08-29 13:50   directory/file4
      591  2020-08-29 13:50   directory/file3
---------                     -------
     3557                     5 files
santi@edith:~/OS/L1$ zip -d dir.zip directory/file4
deleting: directory/file4
santi@edith:~/OS/L1$ unzip -l dir.zip
Archive:  dir.zip
  Length      Date    Time    Name
---------  ---------- -----   ----
        0  2020-08-29 13:42   directory/
      589  2020-08-29 13:50   directory/file2
      893  2020-08-29 13:50   directory/file1
      591  2020-08-29 13:50   directory/file3
---------                     -------
     2073                     4 files
santi@edith:~/OS/L1$
```

e. Unzip the contents from samplezip.zip

```
santi@edith:~/OS/L1$ ls
'50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf'   Input.txt
 directory                                                         linuxcommands.pdf
 directory.zip                                                     sample2.txt
 dir.zip                                                           sample.txt
 empty-dir                                                         samplezip.zip
'Exercise-I LINUX COMMANDS.pdf'
santi@edith:~/OS/L1$ unzip samplezip.zip
Archive:  samplezip.zip
  inflating: sample1.txt
santi@edith:~/OS/L1$ ls
'50_Most_Frequently_Used_UNIX _ Linux Commands_With Examples.pdf'   Input.txt
 directory                                                         linuxcommands.pdf
 directory.zip                                                     sample1.txt
 dir.zip                                                           sample2.txt
 empty-dir                                                         sample.txt
'Exercise-I LINUX COMMANDS.pdf'                                    samplezip.zip
santi@edith:~/OS/L1$
```

End!

```
santi@edith:~$ cowsay -f stegosaurus bye
 _____
< bye >
 -----
\
 \
  \
   \
                                    .       .
                                   / `.   .' "
                           .---.  <    > <    >  .---.
                           |    \  \ - ~ ~ - /  /    |
                         _____          ..-~             ~-..-~
                        |     |   \~~~\.'                    `./~~~/
                       ---------   \__/                        \__/
                      .'  O    \     /                /       \  "
                     (_____,    `._.'                 |         } \/~~~/
                      `----.          /       }      |        /    \__/
                            `-.       |      /       |       /      `. ,~~|
                                ~-.__|      /_ - ~ ^|      /- _      `..-'
                                     |     /        |     /     ~-.     `-. _ _ _
                                     |_____|        |_____|         ~ - . _ _ _ _ _>
santi@edith:~$
```