Q1. Explore the following Commands with various options, refer man pages for

further help

**a) date**

**1**. **--date / -d**

This Command takes date as input string only and displays the output in date format.

**Command :- date --date "8/12/1995"**

**2**. **--file / -f**

If we have file that contains various date strings, then we can use above(--file) Command to outputted each line as date.

**Command :- >cat file\_with\_date**

**>Mar 23 1987**

**>Aug 12 1995**

**>date --file= file\_with\_date**

**3. --set / -s**

We can set time and date of our system using –set option

**Command :- date --set "Sun Feb 7 15:00:00 2013"**

**4. -u, --utc, --universal**

We can display date in UTC format using above Commands.

**Command:-** **date –universal**

**5. date +%a**

Displays weekday name in short like Mon, Tue, Wed, etc.

**6. date +%A**

Displays weekday name in full like Monday, Tuesday, Wednesday, etc.

**7. date +%b**

Displays month name in short like Jan, Feb, Mar etc.

**8. date +%B**

Displays weekday name in full like January, February, March etc

**9. date +%d**

Displays Day of month e.g – 12, 23, etc.

**10. date +%D**

Displays Current Date **MM/DD/YY** format.

**11. date +%F**

Displays Date in YYYY-MM-DD format.

**12. date +%H**

Displays hour in (00..23) format

**13. date +%I**

Displays hour (01..12) format

**14. date +%m**

Displays month in (01..12) format

**15. date +%M**

Displays minute in (00..59) format

**16. date +%S**

Displays second in (00..60) format.

**b) cal**

show calendar

cal may 2020

It is basic calculator in linux. after typing this Command one can do basic calculator operations

c) bc

**d) echo**

echo Command in linux is used to display line of text/string that are passed as an argument .

**1. \b**  : it removes all the spaces in between the text

**2. \c** : suppress trailing new line with backspace interpreter

**3. \n** : this option creates new line from where it is used.

**4. \t :** this option is used to create horizontal tab spaces.

**5. \r :** carriage return with backspace interpreter ‘-e‘ to have specified carriage return in output.

**6. \v :** this option is used to create vertical tab spaces.

**7. \a :** alert return with backspace interpreter ‘-e‘ to have sound alert.

**8. echo \* :** this Command will print all files/folders, similar to ls Command .

**9. -n :** this option is used to omit echoing trailing newline .

**e) Who**

The who Command is used to get information about currently logged in user on to system

**f) whoami**

To display system’s username

**g) logname**

print users login name,Print the name of the current user

**h) uname**

uname is a Command-line utility that prints basic information about the operating system name and system hardware

**-s, (--kernel-name) -** Prints the kernel name.

**-n, (--nodename) –**

Prints the system’s node name (hostname) . This is the name the system uses when communicating over the network. When used with the -n option, uname produces the same output as the hostname Command.

**-r, (--kernel-release) -** Prints the kernel release.

**-v, (--kernel-version) -** Prints the kernel version.

**-m, (--machine) -** Prints the name of the machine’s hardware name.

**-p, (--processor) -** Prints the architecture of the processor.

**-i, (--hardware-platform) -** Prints the hardware platform.

**-o, (--operating-system) -** Print the name of the operating system. On Linux systems that is “GNU/Linux”

**-a, (--all) –**

When the -a option is used, uname behaves the same as if the -snrvmo options have been given.

**h) seq**

seq Command in Linux is used to generate numbers from FIRST to LAST in steps of INCREMENT. It is a very useful Command where we had to generate list of numbers in while, for, until loop.

**i) clear**: this Command is used to clear terminal.

Q2. A) Create three directory a b and c and create a.txt,b.txt and c.txt in each directory respectively and then copy c directory into the a.

**Command: mkdir a b c**

**touch a.txt**

**touch b.txt**

**touch c.txt**

**cp -r c a**

B) Move Directory b to c.

**Command: mv b c**

C) Create alias of ls -lh Command to your name.

**COMMAND:** **alias Sarika=ls**

**alias Sarika=-lh**

D) Change Directory name a to cdac.

**COMMAND:** **$ mv a CDAC**

Q3. A) Create five files file1.txt , file2.txt ,file3.txt file4.txt and

file5.txt with some text inside it. Search for 's' character

inside all the files using grep Command.

$ cat > file1.txt

this is file 1

$ cat > file2.txt

this is file 2

$ cat > file3.txt

this is file 3

$ cat > file4.txt

this is file 4

$ cat > file5.txt

this is file 5

$ grep -o "s" file1.txt

s

s

$ grep -n "s" file1.txt

1: this is file 1

Also Use cat to view all file content together.

B) Create file.txt using cat and edit that using nano editor.

**Cat > file.txt**

**Hello cdac**

c) Create 5 empty files using touch Command

**$ touch f1 f2 f3 f4 f5**

D) Remove previously created directory a , b and c.

**rm -r a**

**rm -r b**

**rm -r c**

Q4. Create a file with some content using vi editor

a) Display the file using cat Command

b) Display line numbers along with content

c) Display the file in reverse order using tac

d) Combine multiple files using cat Command

Q5. Create Random some files and sub directories.

a) display files row wise, column wise, 1 per row

b) sort the files in ascending, descending order

c) Create some hidden files and sub directories and display them using “ls -a”

d) Change time stamp of some files and directories using touch Command and display the files using ls -t ls -rt

e) Recursive display contents of your home directory.

f) display all attributes of files and directories using ls -l

g) display attributes of a directory using ls -ld

==================================================================================

Q1. Create directory of your name and then create 4 files inside it named

a.txt, b.txt, c.txt and d.sh. And change the directory permission to

only user and also files permission to only user such that

a) user can read and write the file a.txt and b.txt

**chmod u=rx a.txt b.txt**

b) user can write the file c.txt

**chmod u+w c.txt**

c) user can exectuable the file d.sh. Make

Contents of file d.sh:

#/bin/sh

ls -lh

Q2. Change the permission of the previous directory to executable only.

Analyze the output.

**chmod +w cdac**

Q3. Change the permission of the previous directory to read only.

Analyze the output.

**chmod +r cdac**

Q4. Use echo to list the all files of your home directory.

**echo .\***

Q5. Use echo to list the all files of /etc/ directory

**$ echo .\* \* /etc /directory**

**$ echo .\* /etc /directory**

Q6. Use echo to list the all files of /etc/ directory with extension name .conf

**$ find /etc -type f -name "\*.conf"**

Q7. Use echo to append your name to a.txt

**echo Sarika >> file3.txt**

==================================================================================

Note: Many operation might not work. Analyze them

1. Create Three level nested directory a, b and c

Command: **touch a🡪Cd a🡪 Mkdir b🡪touch .a.txt🡪touch a.txt**

**Cd b🡪 Mkdir b🡪touch .b.txt🡪touch b.txt**

**Cd c🡪 Mkdir c🡪touch .c.txt🡪touch c.txt**

a) Create at least one hidden file in each directory

**COMMAND🡪 touch a🡪Cd a🡪 Mkdir b🡪touch .a.txt**

**Cd b🡪 Mkdir b🡪touch .b.txt**

**Cd c🡪 Mkdir c🡪touch .c.txt**

b) Create at least one normal text file in each directory

**touch a🡪Cd a🡪 Mkdir b🡪touch a.txt**

**Cd b🡪 Mkdir b🡪touch b.txt**

**Cd c🡪 Mkdir c🡪 touch c.txt**

c) Copy c into a

**cp -r c a**

d) Copy file inside a into b

1. Change the third level directory c permission to read only

**COMMAND🡪 chmod +r c**

1. Try copying third level c directory to outside of a

**COMMAND** 🡪**cp –r cc bb**

1. Create the directory “data“ inside b

**COMMAND** 🡪**cd b 🡪mkdir data**

1. Change the permission of b to read and write only

**COMMAND** 🡪**chmod +r+w b**

1. Create directory “data2” inside b

**COMMAND** 🡪**cd b🡪mkdir data2**

1. Create data.txt file inside b

**COMMAND** 🡪**cd b🡪touch data.txt**

1. Rename the second level directory c to copied\_dir

**COMMAND** **🡪mv c copied\_dir**

1. Change permission of copied\_dir to only executable

**COMMAND** **🡪chmod +w copied\_dir**

1. Try ls on copied\_dir

**COMMAND -> ls**

1. Try moving copied\_dir to outside of a

**COMMAND** **🡪 mv copied\_dir bb**

1. Try moving data directory outside of a

**COMMAND** **🡪 mv data bb**

1. Now change the owner of the directory a to root
2. Change permission of a to read write and executable to only user

**COMMAND** **🡪 Chmod u+r+w+x aa**

1. Try creating directory sample inside a

**COMMAND** **🡪 Mkdir sample**

1. Create directory x outside a

**COMMAND** **🡪 Mkdir y**

1. Try moving x inside a

**COMMAND** **🡪 Mv x a**

1. Create directory movies

**COMMAND** **🡪 Mkdir movies**

1. Change group of movies to sudo