JAIN
DEEMED-TO-BE UNIVERSITY

SCHOOL OF COMPUTER SCIENCE AND IT

DEPARTMENT OF BACHELOR OF COMPUTER APPLICATIONS

Introduction to Linux

Lab Manual

Subject Code: 20BCA5D21

Class: I Year I Semester

Prepared By:

Suman Garai

JU2020BCAS19059

Faculty In-Charge:

Ananta Ojha

&

Akhil Menon

Student Signature:

Faculty Signature:

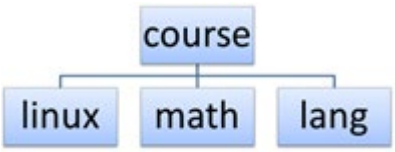
Lab Experiment 1: Execute the commands based on the given problem statements, referring to creating directories and file structure.

⇒ When you are at your home directory, give command to display a long list of all files and directories including hidden ones present in '/usr' directory sorted based on their modification time.

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ ls -alt /usr/
total 124
drwxr-xr-x   2 root root 20480 Dec 13 19:30 sbin
drwxr-xr-x   2 root root 40960 Dec 11 09:55 bin
drwxr-xr-x   6 root root  4096 Dec 11 09:55 src
drwxr-xr-x  12 root root 12288 Dec  4 10:01 libexec
drwxr-xr-x 116 root root  4096 Nov 21 11:04 lib
drwxr-xr-x  10 root root  4096 Nov 21 11:03 include
drwxr-xr-x   2 root root  4096 Nov 21 11:02 lib64
drwxr-xr-x 249 root root 12288 Nov  5 23:13 share
drwxr-xr-x  20 root root  4096 Nov  5 23:10 ..
drwxr-xr-x   2 root root  4096 Apr 23  2020 games
drwxr-xr-x  14 root root  4096 Apr 23  2020 .
drwxr-xr-x  10 root root  4096 Apr 23  2020 local
drwxr-xr-x   2 root root  4096 Apr 23  2020 lib32
drwxr-xr-x   2 root root  4096 Apr 23  2020 libx32
gboi3122@biist-VirtualBox:~/Desktop$
```

⇒ Create a directory structure as given in Figure-1 in your home directory.



Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ mkdir -p course/{linux,lang,math}
gboi3122@biist-VirtualBox:~/Desktop$ ls -R course/
course/:
lang  linux  math

course/lang:

course/linux:

course/math:
gboi3122@biist-VirtualBox:~/Desktop$
```

⇒ Change your working directory to ‘math’ and when you are in ‘math’ give a command to display all files and directories including hidden ones present in home directory using a multi-column output.

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ cd course/math/
gboi3122@biist-VirtualBox:~/Desktop/course/math$ ls -alx ~
.                       ..          .bash_history              .bash_logout                    .bashrc
.bashrc.swp             .cache      .config                    .dbus                           Desktop
Documents               Downloads   .gnupg                     .local                          .mozilla
Music                   Pictures    .profile                   Public                          snap
.sudo_as_admin_successful Templates .vboxclient-clipboard.pid  .vboxclient-display-svga-x11.pid .vboxclient-draganddrop.pid
.vboxclient-seamless.pid Videos     .viminfo
gboi3122@biist-VirtualBox:~/Desktop/course/math$
```

⇒ Change your working directory to ‘linux’, create a file ‘first.sh’ in it. Display current logged-in users and their login time.

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop/course/math$ cd ~/Desktop/course/linux
gboi3122@biist-VirtualBox:~/Desktop/course/linux$ vi first.sh
gboi3122@biist-VirtualBox:~/Desktop/course/linux$ who
gboi3122 :0           2020-12-13 19:30 (:0)
gboi3122@biist-VirtualBox:~/Desktop/course/linux$
```

⇒ When you are at ‘linux’ directory, create a directory structure as given in Figure-2 inside ‘math’ directory.

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop/course/linux$ mkdir -p ~/course/math/sets/fuzzy/rough
gboi3122@biist-VirtualBox:~/Desktop/course/linux$
gboi3122@biist-VirtualBox:~/Desktop/course/linux$ ls -R ~/course/math/
/home/gboi3122/course/math/:
sets

/home/gboi3122/course/math/sets:
fuzzy

/home/gboi3122/course/math/sets/fuzzy:
rough

/home/gboi3122/course/math/sets/fuzzy/rough:
gboi3122@biist-VirtualBox:~/Desktop/course/linux$
```

⇒ Change your working directory to ‘math’ and remove all the three directories ‘sets’, ‘fuzzy’ and ‘rough’ using a single command.

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop/course/linux$ cd ../math/
gboi3122@biist-VirtualBox:~/Desktop/course/math$ rm -rf sets/
gboi3122@biist-VirtualBox:~/Desktop/course/math$
```

Lab Experiment 2 : Execute the Commands based on creating files with pipes, wildcards using grep, sed, awk, commands.

Use a text editor of your choice to create a file 'sales.dat' containing the following sales data on a product. Do not include the header in the data file:

Product	Month	Quantity	Amount (Rs.)
Refrigerator	June	18	244500
Washing Machine	August	17	219500
Television	July	20	314000
Washing Machine	September	21	325700
Refrigerator	September	17	210000
Television	October	25	450000

```
gboi3122@biist-VirtualBox:~/Desktop$ cat > sales.dat
refrigerator june 18 244500
washingmachine august 17 219500
television july 20 314000
washingmachine september 21 325700
refrigerator september 17 210000
television october 25 450000
```

Based on the above data file, answer any two of the following questions:

⇒ Using grep command, display sales of Washing Machine.

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ grep 'washingmachine' sales.dat
washingmachine august 17 219500
washingmachine september 21 325700
```

⇒ Using sed command, append the following data in the file.
Refrigerator October 22 320000

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ sed '6a\refrigerator october 22 320000' sales.dat
refrigerator june 18 244500
washingmachine august 17 219500
television july 20 314000
washingmachine september 21 325700
refrigerator september 17 210000
television october 25 450000
refrigerator october 22 320000
```

⇒ Using awk command, display sales of July and September months.

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ awk '$2=="july" || $2=="september"{print $0}' sales.dat
television july 20 314000
washingmachine september 21 325700
refrigerator september 17 210000
```

⇒ Using grep command, display sales of all products except Television.

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ grep -v 'television' sales.dat
refrigerator june 18 244500
washingmachine august 17 219500
washingmachine september 21 325700
refrigerator september 17 210000
```

Lab Experiment 3 : Create a data file called 'employee' in the format given below:

a. EmpCode Character
b. EmpName Character
c. Grade Character
d. Years of experience Numeric
e. Basic Pay Numeric

For example:

A001	ARJUN	E1	01	12000.00
A006	Anand	E1	01	12450.00
A010	Rajesh	E2	03	14500.00
A002	Mohan	E2	02	13000.00
A005	John	E2	01	14500.00
A009	Denial Smith	E2	04	17500.00
A004	Williams	E1	01	12000.00

```
A001    ARJUN           E1      01      12000.00
A006    Anand           E1      01      12450.00
A010    Rajesh          E2      03      14500.00
A002    Mohan           E2      02      13000.00
A005    John            E2      01      14500.00
A009    Denial Smith    E2      04      17500.00
A004    Williams        E1      01      12000.00
```

Perform the following operations on the file:

⇒ Sort the file on EmpCode.

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ sort -k1 employee.dat | tr '_' ' '
A001    ARJUN           E1      01      12000.00
A002    Mohan           E2      02      13000.00
A004    Williams        E1      01      12000.00
A005    John            E2      01      14500.00
A006    Anand           E1      01      12450.00
A009    Denial Smith    E2      04      17500.00
A010    Rajesh          E2      03      14500.00
```

⇒ Sort the file on: - Decreasing order of basic pay

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ sort -k5 -r employee.dat | tr '_' ' '
A009    Denial Smith    E2      04      17500.00
A010    Rajesh          E2      03      14500.00
A005    John            E2      01      14500.00
A002    Mohan           E2      02      13000.00
A006    Anand           E1      01      12450.00
A004    Williams        E1      01      12000.00
A001    ARJUN           E1      01      12000.00
```

⇒ Sort the file on: - Increasing order of years of experience

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ sort -k4 employee.dat | tr '_' ' '
A001    ARJUN           E1      01      12000.00
A004    Williams        E1      01      12000.00
A006    Anand           E1      01      12450.00
A005    John            E2      01      14500.00
A002    Mohan           E2      02      13000.00
A010    Rajesh          E2      03      14500.00
A009    Denial Smith    E2      04      17500.00
```

⇒ Display the number of employees whose details are included in the file.

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ echo -n "Employee = "; cat employee.dat | wc -l
Employee = 7
```

⇒ Display all records with ‘smith’ a part of the employee name.

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ awk '/Smith/ {print $0}' employee.dat | tr '_' ' '
A009    Denial Smith    E2      04      17500.00
```

⇒ Display all records with EmpName starting with ‘B’.

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ awk '/[B]/ {print $0}' employee.dat
gboi3122@biist-VirtualBox:~/Desktop$
```

⇒ Display the records on employees whose grade is E2 and have work experience of 2 to 5 years.

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ awk '$3=="E2" && ($4>=02 && $4<=5) {print $0}' employee.dat | tr '_' ' '
A010    Rajesh          E2      03      14500.00
A002    Mohan           E2      02      13000.00
A009    Denial Smith    E2      04      17500.00
```

⇒ Save the names of all employees whose basic pay is between 10000 and 15000 in a file ‘employee1’

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ awk '$5>=10000.00 && $5<=15000.00 {print $0}' employee.dat
A001    ARJUN           E1      01      12000.00
A006    Anand           E1      01      12450.00
A010    Rajesh          E2      03      14500.00
A002    Mohan           E2      02      13000.00
A005    John            E2      01      14500.00
A004    Williams        E1      01      12000.00
```

⇒ Display records of all employees who are not in grade E2.

Code & Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ awk '$3!="E2" {print $0}' employee.dat
A001    ARJUN           E1      01      12000.00
A006    Anand           E1      01      12450.00
A004    Williams        E1      01      12000.00
```

Lab Experiment 4 : WAP that accepts user name and reports if the user is logged in

Code:

```
#!/bin/shh
read -p "Enter Username:" u
who > userlist
if grep $u userlist
then
        echo "The User is Logged In"
else
        echo "The User has'nt Logged In"
fi
```

Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ vi lexp4.sh
gboi3122@biist-VirtualBox:~/Desktop$ . lexp4.sh
Enter Username:gboi3122
gboi3122 :0           2020-12-13 19:30 (:0)
The User is Logged In
gboi3122@biist-VirtualBox:~/Desktop$ . lexp4.sh
Enter Username:acojha
The User has'nt Logged In
gboi3122@biist-VirtualBox:~/Desktop$
```

Lab Experiment 5 : WAP to take a file as a command-line argument and reverse the contents of the file and save it in a new file.

Code:

```
#!/bin/sh
if [ -e $1 ]
then
        echo " Contents of the File $1: "
        cat $1
        echo " Reversing the File: "
        tac $1
        echo "Saving the Reversed File as a New File ..."
        tac $1 | cat > reverse.txt
else
        echo "The File doesn't Exist"
fi
```

Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ vi lexp5.sh
gboi3122@biist-VirtualBox:~/Desktop$ . lexp5.sh employee.dat
 Contents of the File employee.dat:
A001    ARJUN           E1      01      12000.00
A006    Anand           E1      01      12450.00
A010    Rajesh          E2      03      14500.00
A002    Mohan           E2      02      13000.00
A005    John            E2      01      14500.00
A009    Denial_Smith    E2      04      17500.00
A004    Williams        E1      01      12000.00
 Reversing the File:
A004    Williams        E1      01      12000.00
A009    Denial_Smith    E2      04      17500.00
A005    John            E2      01      14500.00
A002    Mohan           E2      02      13000.00
A010    Rajesh          E2      03      14500.00
A006    Anand           E1      01      12450.00
A001    ARJUN           E1      01      12000.00
Saving the Reversed File as a New File ...
gboi3122@biist-VirtualBox:~/Desktop$
```

Lab Experiment 6 : WAP that takes a filename as input and checks if it is executable, if not make it executable.

Code:

```
#!/bin/bash
echo "Ennter the Filename: "
read file
if [ -e $file ]
then
        if [ -x $file ]
        then
                echo "The File Is Executable"
                ls -l $file
        else
                ls -l $file
                echo "File is not Executable, Changing Permissions ..."
                chmod +x $file
                echo "The File's Permissions has been Changed."
                ls -l $file
        fi
else
        echo "The File Doesn't Exist."
fi
```

Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ vi lexp6.sh
gboi3122@biist-VirtualBox:~/Desktop$ . lexp6.sh
Ennter the Filename:
lexp5.sh
-rw-rw-r-- 1 gboi3122 gboi3122 229 Dec 13 23:42 lexp5.sh
File is not Executable, Changing Permissions ...
The File's Permissions has been Changed.
-rwxrwxr-x 1 gboi3122 gboi3122 229 Dec 13 23:42 lexp5.sh
gboi3122@biist-VirtualBox:~/Desktop$ . lexp6.sh
Ennter the Filename:
course
The File Is Executable
total 12
drwxrwxr-x 2 gboi3122 gboi3122 4096 Dec 13 19:33 lang
drwxrwxr-x 2 gboi3122 gboi3122 4096 Dec 13 19:44 linux
drwxrwxr-x 2 gboi3122 gboi3122 4096 Dec 13 19:33 math
gboi3122@biist-VirtualBox:~/Desktop$
```

Lab Experiment 7 : WAP that takes a filename as input and checks if it is a directory file, if so then lists its contents.

Code:

```
#!/bin/sh
echo "Enter the Filename: "
read file
if [ -e $file ]
then
        if [ -d $file ]
        then
                echo "The File is Directory"
                echo " Contents of the File are: "
                ls -R $file
        else
                echo "The File isn't Directory."
        fi
else
        echo "The File doesn't exist."
fi
```

Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ vi lexp7.sh
gboi3122@biist-VirtualBox:~/Desktop$ . lexp7.sh
Enter the Filename:
course
The File is Directory
 Contents of the File are:
course:
lang  linux  math

course/lang:

course/linux:
first.sh

course/math:
gboi3122@biist-VirtualBox:~/Desktop$ . lexp7.sh
Enter the Filename:
lexp6.sh
The File isn't Directory.
gboi3122@biist-VirtualBox:~/Desktop$ . lexp7.sh
Enter the Filename:
acojha
The File doesn't exist.
gboi3122@biist-VirtualBox:~/Desktop$
```

Lab Experiment 8 : WAP that replaces all ".txt" file names with ".txt.old" in the current directory.

Code:

```
#!/bin/sh
for file in *.txt
do
        mv $file $file.old
done
```

Output:

```
gboi3122@biist-VirtualBox:~/Desktop$ touch file1.txt file2.txt file3.txt file4.txt file5.txt
gboi3122@biist-VirtualBox:~/Desktop$ ls *.txt
file1.txt  file2.txt  file3.txt  file4.txt  file5.txt
gboi3122@biist-VirtualBox:~/Desktop$ . lexp8.sh
gboi3122@biist-VirtualBox:~/Desktop$ ls *txt*
file1.txt.old  file2.txt.old  file3.txt.old  file4.txt.old  file5.txt.old  txt.old
gboi3122@biist-VirtualBox:~/Desktop$
```

Lab Experiment 9 : WAP which displays the following menu and executes the option selected by the user: - 1. ls -l 2. pwd 3. ps -fe

Code:

```
#1/bin/bash
while true
do
        clear
        echo "MENU"
        echo "1. List"
        echo "2. pwd"
        echo "3. Show all processes"
        echo "0. Exit"
        read -p "Enter Option: " opt
        case $opt in
                1) ls -l
                        sleep 10
                        ;;
                2) pwd
                        sleep 10
                        ;;
                3) ps -fe
                        sleep 10
                        ;;
                0) echo "Quitting the menu"
                        sleep 3
                        break
                        ;;
                *) echo "Invalid Input"
                        sleep 3
        esac
done
```

Output:

```
MENU
1. List
2. pwd
3. Show all processes
0. Exit
Enter Option: 1
total 24
drwxrwxr-x 5 gboi3122 gboi3122 4096 Dec 13 19:33 course
-rwxrwxr-x 1 gboi3122 gboi3122  229 Dec 13 23:42 lexp5.sh
-rw-rw-r-- 1 gboi3122 gboi3122  345 Dec 14 00:11 lexp6.sh
-rw-rw-r-- 1 gboi3122 gboi3122  260 Dec 14 00:21 lexp7.sh
-rw-rw-r-- 1 gboi3122 gboi3122   97 Dec 14 01:07 lexp8.sh
-rw-rw-r-- 1 gboi3122 gboi3122  358 Dec 14 02:07 lexp9.sh
-rw-rw-r-- 1 gboi3122 gboi3122    0 Dec 14 01:40 txt.old
```

```
MENU
1. List
2. pwd
3. Show all processes
0. Exit
Enter Option: 2
/home/gboi3122/Desktop
```

```
MENU
1. List
2. pwd
3. Show all processes
0. Exit
Enter Option: 3
UID          PID    PPID  C STIME TTY          TIME CMD
root           1       0  0 00:39 ?        00:00:01 /sbin/init splash
root           2       0  0 00:39 ?        00:00:00 [kthreadd]
root           3       2  0 00:39 ?        00:00:00 [rcu_gp]
root           4       2  0 00:39 ?        00:00:00 [rcu_par_gp]
root           6       2  0 00:39 ?        00:00:00 [kworker/0:0H-kblockd]
root           9       2  0 00:39 ?        00:00:00 [mm_percpu_wq]
root          10       2  0 00:39 ?        00:00:00 [ksoftirqd/0]
root          11       2  0 00:39 ?        00:00:01 [rcu_sched]
root          12       2  0 00:39 ?        00:00:00 [migration/0]
root          13       2  0 00:39 ?        00:00:00 [idle_inject/0]
root          14       2  0 00:39 ?        00:00:00 [cpuhp/0]
root          15       2  0 00:39 ?        00:00:00 [cpuhp/1]
root          16       2  0 00:39 ?        00:00:00 [idle_inject/1]
root          17       2  0 00:39 ?        00:00:00 [migration/1]
root          18       2  0 00:39 ?        00:00:00 [ksoftirqd/1]
root          20       2  0 00:39 ?        00:00:00 [kworker/1:0H-kblockd]
root          21       2  0 00:39 ?        00:00:00 [cpuhp/2]
root          22       2  0 00:39 ?        00:00:00 [idle_inject/2]
root          23       2  0 00:39 ?        00:00:00 [migration/2]
root          24       2  0 00:39 ?        00:00:00 [ksoftirqd/2]
root          26       2  0 00:39 ?        00:00:00 [kworker/2:0H-kblockd]
root          27       2  0 00:39 ?        00:00:00 [cpuhp/3]
root          28       2  0 00:39 ?        00:00:00 [idle_inject/3]
root          29       2  0 00:39 ?        00:00:00 [migration/3]
root          30       2  0 00:39 ?        00:00:00 [ksoftirqd/3]
root          32       2  0 00:39 ?        00:00:00 [kworker/3:0H-kblockd]
root          33       2  0 00:39 ?        00:00:00 [cpuhp/4]
root          34       2  0 00:39 ?        00:00:00 [idle_inject/4]
root          35       2  0 00:39 ?        00:00:00 [migration/4]
root          36       2  0 00:39 ?        00:00:00 [ksoftirqd/4]
root          38       2  0 00:39 ?        00:00:00 [kworker/4:0H-kblockd]
root          39       2  0 00:39 ?        00:00:00 [cpuhp/5]
root          40       2  0 00:39 ?        00:00:00 [idle_inject/5]
root          41       2  0 00:39 ?        00:00:00 [migration/5]
root          42       2  0 00:39 ?        00:00:00 [ksoftirqd/5]
root          44       2  0 00:39 ?        00:00:00 [kworker/5:0H-kblockd]
root          45       2  0 00:39 ?        00:00:00 [kdevtmpfs]
root          46       2  0 00:39 ?        00:00:00 [netns]
root          47       2  0 00:39 ?        00:00:00 [rcu_tasks_kthre]
```

```
MENU
1. List
2. pwd
3. Show all processes
0. Exit
Enter Option: 0
Quitting the menu
gboi3122@biist-VirtualBox:~/Desktop$
```

Lab Experiment 10 : WAP to run a shell script, when the system starts up

⇒ To set reminders about daily routine for a user

Code:

```
#!/bin/bash
echo "<<------------ ROUTINE ------------>>"
echo "  Good Morning"
echo "  Breakfast before 9:30"
echo "  Classes from 9:30 to 3"
echo "  Lunch + Freetime till 7"
echo "  Study from 7"
echo "  Go to Bed before 3"
```

```
gboi3122@biist-VirtualBox:~/Desktop$ vi lexp10.sh
gboi3122@biist-VirtualBox:~/Desktop$ chmod +x lexp10.sh
```

```
    #alias dir='dir --color=auto'
    #alias vdir='vdir --color=auto'

    alias grep='grep --color=auto'
    alias fgrep='fgrep --color=auto'
    alias egrep='egrep --color=auto'
fi

# colored GCC warnings and errors
#export GCC_COLORS='error=01;31:warning=01;35:note=01;36:caret=01;32:locus=01:quote=01'

# some more ls aliases
alias ll='ls -alF'
alias la='ls -A'
alias l='ls -CF'

# Add an "alert" alias for long running commands.  Use like so:
#   sleep 10; alert
alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo terminal || echo error)" "$(history|tail -n1|sed -e '\''s/^\s*[0-9]\+\s*//;s/[;&|]\s*alert$//'\'')"'

# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

~/Desktop/lexp10.sh
```

Output:

```
<<------------ ROUTINE ------------>>
  Good Morning
  Breakfast before 9:30
  Classes from 9:30 to 3
  Lunch + Freetime till 7
  Study from 7
  Go to Bed before 3
gboi3122@biist-VirtualBox:~/Desktop$
```

⇒ To sort the documents in the downloads folder, based on the file extensions

Code:

```
#!/bin/bash
ls -x --sort=extension ~/Downloads
```

Output:

```
gbot3122@blist-VirtualBox:~/Desktop$ . lexp10.1.sh
course    file1.txt.old  file2.txt.old  file3.txt.old  file4.txt.old  file5.txt.old  txt.old  lexp10.1.sh  lexp10.sh  lexp5.sh  lexp6.sh  lexp7.sh
lexp8.sh  lexp9.sh
```

-- THE END --