Operating Systems Lab

Semester-5, B.tech(ICT)

Roll No:-1401010, Name:-Deep C. Patel

Assignment-1

1).Script -1

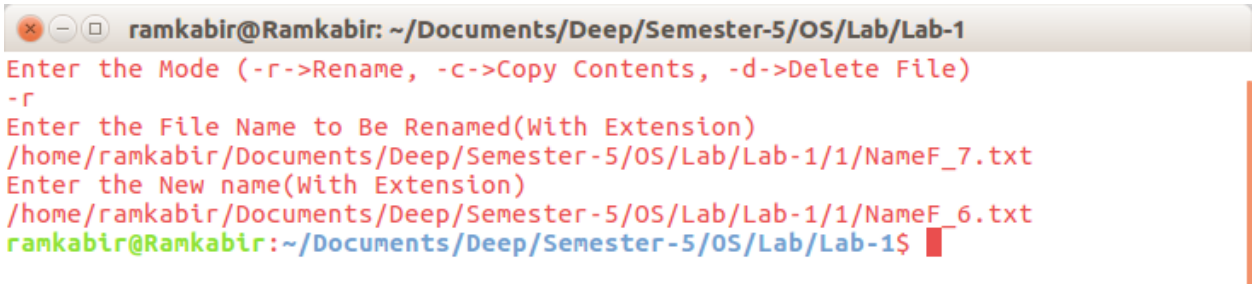
Write a script to obtain the effect DELETE/CONFIRM command. Generalize it to be used for COPY/CONFIRM and RENAME/CONFIRM.

Code:-

```
#!/bin/bash

#Created By:-Deep C. Patel, 1401010

clear

echo "Enter the Mode (-r->Rename, -c->Copy Contents, -d->Delete
File)"
read v1

if [ $v1 = "-r" ]
then
    echo "Enter the File Name to Be Renamed(With Extension)"
    read v2

    echo "Enter the New name(With Extension)"
    read v3

    if [ -e $v2 ]
    then
        if [ $v3 ]
        then
            mv -i $v2 $v3
        else
            echo "Less Arguments than Expected"
        fi
    else
        echo "Path Not Found..."
```

```
    fi

elif [ $v1 = "-d" ]
then

    echo "Enter the File Name to Be Deleted(With Extension)"
    read v2

    if [ -e $v2 ]
    then
        rm -i $v2
    else
        echo "Path Not Found..."
    fi

elif [ $v1 = "-c" ]
then
    echo "Enter the Host File Name(With Extension)"
    read v2

    echo "Enter the Destination File Name(With Extension)"
    read v3

    if [ -e $v2 ]
    then
        if [ $v3 ]
        then
            cp -Ti $v2 $v3
        else
            echo "Less Arguments than Expected"
        fi
    else
        echo "Path Not Found..."
    fi

else
    echo "Error:Syntax Wrong.."
fi
```

Screenshot:-



Note: - This code Renames, Copies, Removes the file specified, here the screen shot is of renaming the file NameF_7.txt to NameF_6.txt, which it performs successfully.

---

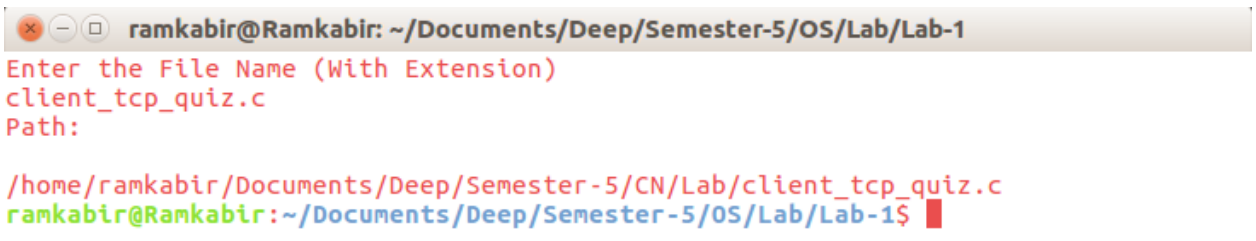
2).Script -3

Input a file name from a user and find out the complete path for a give file name.

Code:-

```
#!/bin/bash

#Created By:-Deep C. Patel, 1401010

clear

echo "Enter the File Name (With Extension)"
read v1

echo -e "Path:\n"

find "/home/ramkabir/Documents/" -type f -name $v1
```

Screenshot:-



---

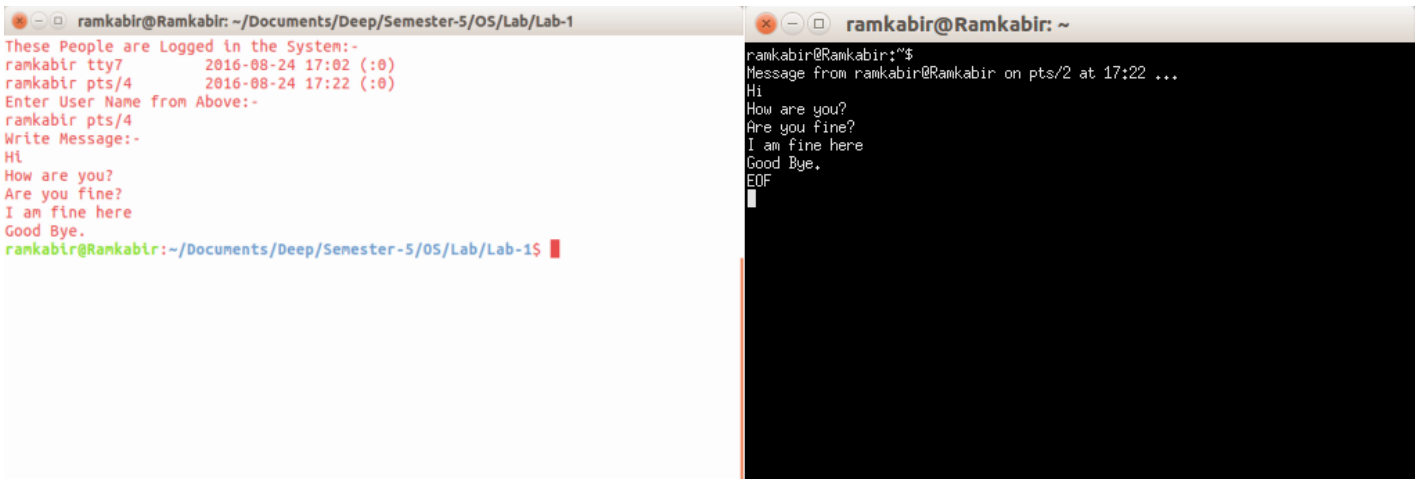
3).Script-4

Write a script to broadcast a message to a specified user or a group of users logged on any terminal.

Code:-

```
#!/bin/bash

#Created By:-Deep C. Patel, 1401010

clear

echo "These People are Logged in the System:-"
who

echo "Enter User Name from Above:-"
read user_name

echo "Write Message:-"

write $user_name
```

Screenshot:-



4).Script-5

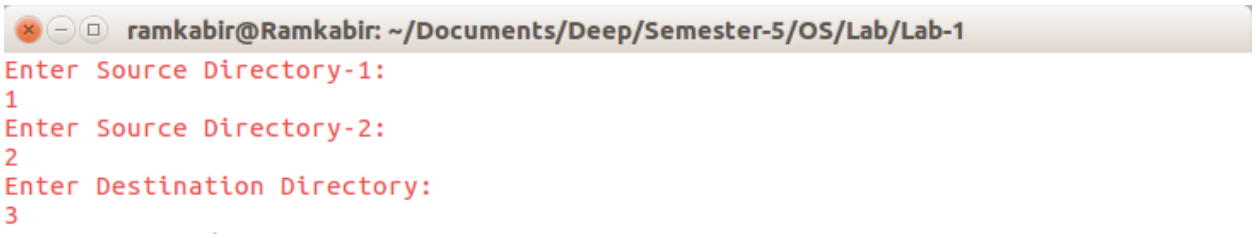
Write a script to copy the files from two directories onto a new directory in such a way that only the latest file is copied, in case there are common files in both the directories.

Code:-

```
#!/bin/bash

#Created by Deep C. Patel-1401010

clear

echo "Enter Source Directory-1:"
read dir1

echo "Enter Source Directory-2:"
read dir2

echo "Enter Destination Directory:"
read dir3

for f in $dir1/*                          #Iterating for files in
folder
do
    f12=$( basename "/$f" )                      #Extracting file
name from path

    if [ -f "$dir2/$f12" ]                  #Checking whether
path(file path) exists or not
    then

        if [ "$dir1/$f12" -nt "$dir2/$f12" ]        #Comparing
the modification date of files
        then
             cp "$dir1/$f12" "$dir3"
        else
             cp "$dir2/$f12" "$dir3"
        fi

    else
        cp "$dir1/$f12" "$dir3"

    fi

done
```

```
for f in $dir2/*
do

    cp -n $f "$dir3"                          #Copying the files
and not replacing existing files in $dir3

done
```

Screenshot:-



Note: - Here the script successfully copies the latest contents from both the directories(1 and 2) to the third(3) directory. We can also give full path of the directories. Here in screenshot we have used relative path.

---

5).Script-6

Write a script to display the files in the specified directory in the following format: File Size in KB Date Protection Owner At the end display total number of files occupying total space.

Code:-

```
#!/bin/bash

#Created by Deep C. Patel-1401010

clear

echo "Enter the Directory:"
read dir1
count=0

path=$( find $dir1 -print )
```

```
for f in $path                                  #Iterating for files
in folder
do
    set -- `ls -l --block-size=K "$f"`

    if [ $1 != "total" ]
    then

        echo "$9 $5 $6 $7 $8 $1 $3"
        count=$((count+1))
    fi

done

echo -e "\nTotal Files: $count"
echo "Total Size:"
echo `du -sh $dir1`
```

Screenshot:-

```
./poem_2.txt 1K Aug 22 17:16 -rw-rw-r-- ramkabir
./Script_1_19.sh 1K Aug 22 01:18 -rw-rw-r-- ramkabir
./Script_1_6.sh 1K Aug 24 18:58 -rw-rw-r-- ramkabir
./Script_1_31.sh 1K Aug 22 15:04 -rw-rw-r-- ramkabir
./Script_1_22.sh 1K Aug 22 01:18 -rw-rw-r-- ramkabir
./Screenshots/Image_01.png 353K Aug 24 15:17 -rw-rw-r-- ramkabir
./Screenshots/Image_03.png 332K Aug 24 15:39 -rw-rw-r-- ramkabir
./Screenshots/Image_05.png 288K Aug 24 18:30 -rw-rw-r-- ramkabir
./Screenshots/Image_04.png 1364K Aug 24 17:27 -rw-rw-r-- ramkabir

Total Files: 66
Total Size:
2.6M ./
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-1$
```

---

6).Script-8

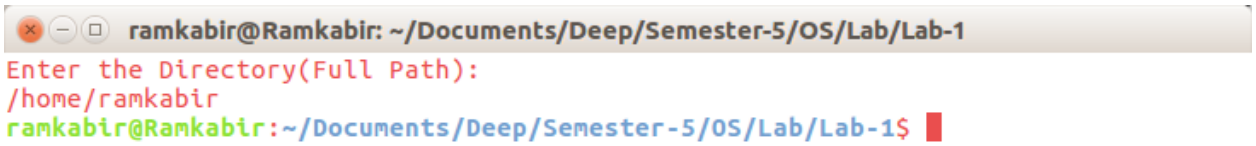
Write a script to delete zero sized files from a given directory (and all its sub directories).

Code:-

```
#!/bin/bash

#Created by Deep C. Patel-1401010
```

```
clear

echo "Enter the Directory(Full Path):"
read dir1

iterate=$( find $dir1 -size 0 -print )

for f in $iterate                                   #Iterating for
files in folder
do
    rm $f

done
```

Screenshot:-



Note: - The code successfully deletes the files of 0 size, it uses find command for it.

7).Script-10

Write a script to display the name of all executable files in the given directory.

Code:-

```
#!/bin/bash

#Created by Deep C. Patel-1401010

clear

echo "Enter the Directory:"
read dir1

iterate=$( find $dir1 -executable -type f )             #Finding
files of specific type and storing its path in iterate
                                                        #-type f
suggests regular file, -executable suggest find executable files

for f in $iterate
#Iterating for files in folder
do
    echo "Executable File is: ` basename "/$f" `"

done
```

Screenshot:-

8).Script-11

Write a script to display the date, time and a welcome message (like Good Morning etc.) The time should be displayed with “a.m.” Or “p.m.” and not in terms of 24 hours notation.

Code:-

```
#!/bin/bash

#Created by Deep C. Patel-1401010

clear

echo "Date: `date -I`"
echo "Time:`date "+ %I:%M:%S"`"

if [ `date "+ %p"` = "AM" ]
then

    echo -e "\nGood Morning........."

else

    if [ `date '+ %H'` -ge 20 ]
    then

        echo -e "\nGood Night........."

    elif [ `date '+ %H'` -ge 17 ]
    then

        echo -e "\nGood Evening........."

    elif [ `date '+ %H'` -gt 12 ]
    then

        echo -e "\nGood After Noon........."

    fi

fi
```

Screenshot:-

---

9).Script-12

Write a script to display the directory in the descending order of the size of each file.

Code:-

```
#!/bin/bash

#Created by Deep C. Patel-1401010

clear

echo "Enter the Directory:"
read dir1

path=$( find $dir1 -type f )

for f in `ls -S $path`
```

```
do
    echo `basename $f`
done
```

Screenshot:-

---

10).Script-15

Write a script which reads a text file and output the following:

a. Count of characters, words and lines

b. File in a reversed order

c. Frequency of particular word in the file

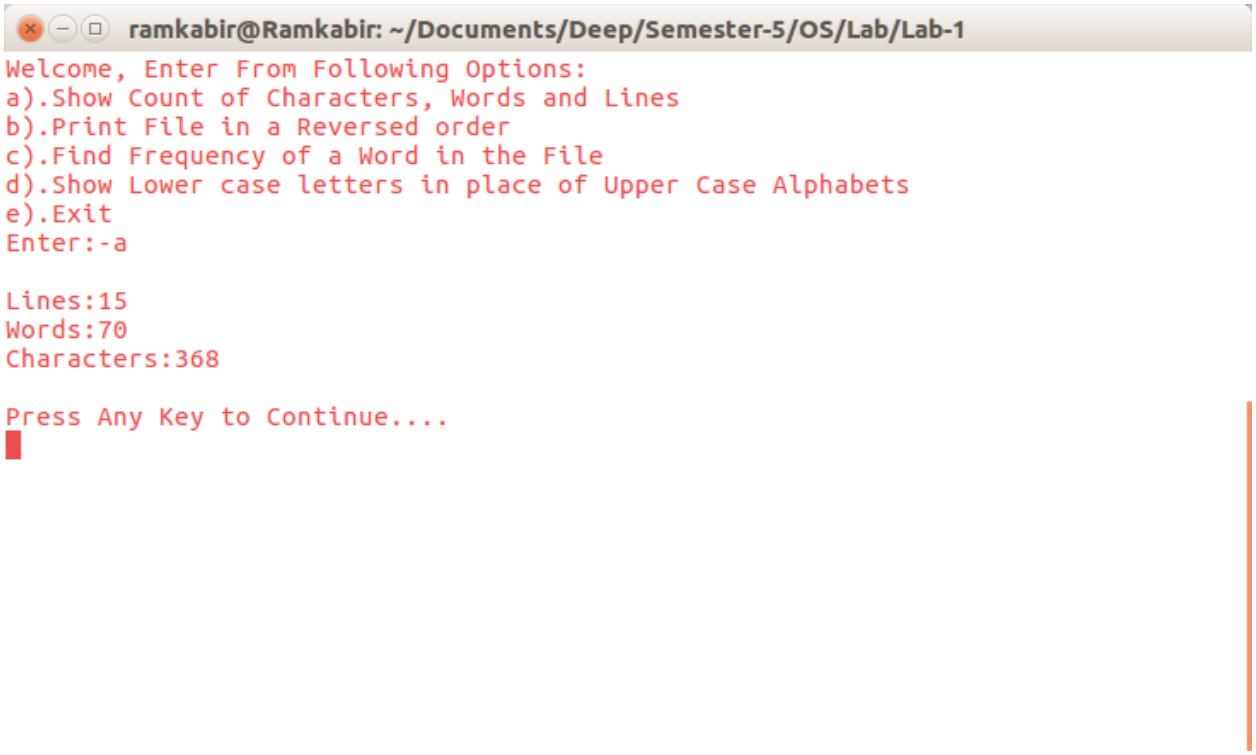
d. Lower case letters in place of upper case alphabets

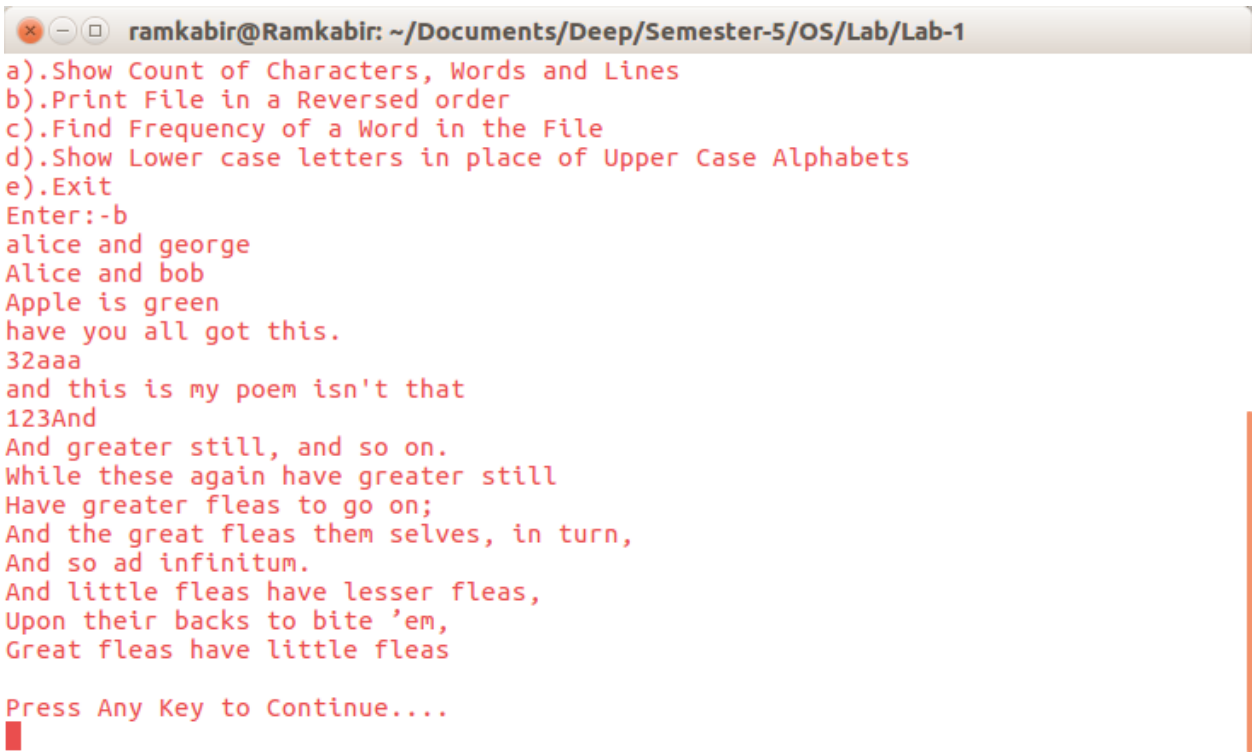
Code:-

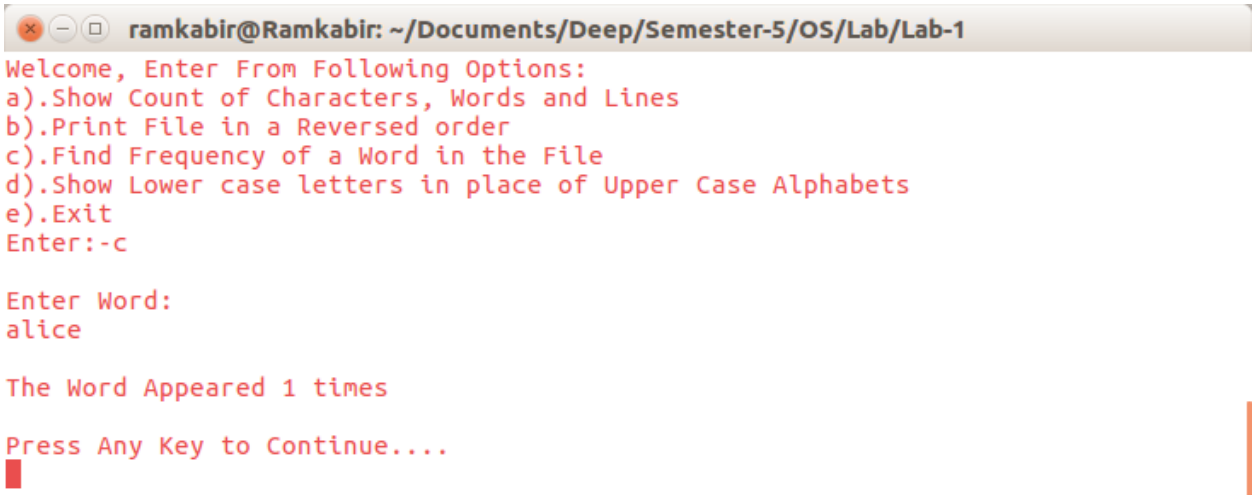
```
#!/bin/bash

#Created by Deep C. Patel-1401010

clear

echo "Enter the file Name(Full Path with Extension):"
read file1

while [ 1 ]
do
    clear

    echo "Welcome, Enter From Following Options:"
    echo "a).Show Count of Characters, Words and Lines"
    echo "b).Print File in a Reversed order"
    echo "c).Find Frequency of a Word in the File"
    echo "d).Show Lower case letters in place of Upper Case
Alphabets"
    echo "e).Exit"
    echo -e "Enter:-\c"
    read ans

    case $ans in

        [aA])   echo -e "\nLines:`wc -l < $file1`"
                echo "Words:`wc -w < $file1`"
                echo "Characters:`wc -m < $file1`"

                echo -e "\nPress Any Key to Continue...."
                read v

                continue;;

        [bB])   tac $file1 > "temp.txt"
                cat "temp.txt"
                rm "temp.txt"

                echo -e "\nPress Any Key to Continue...."
                read v

                continue;;

        [cC])   echo -e "\nEnter Word:"
                read word
```

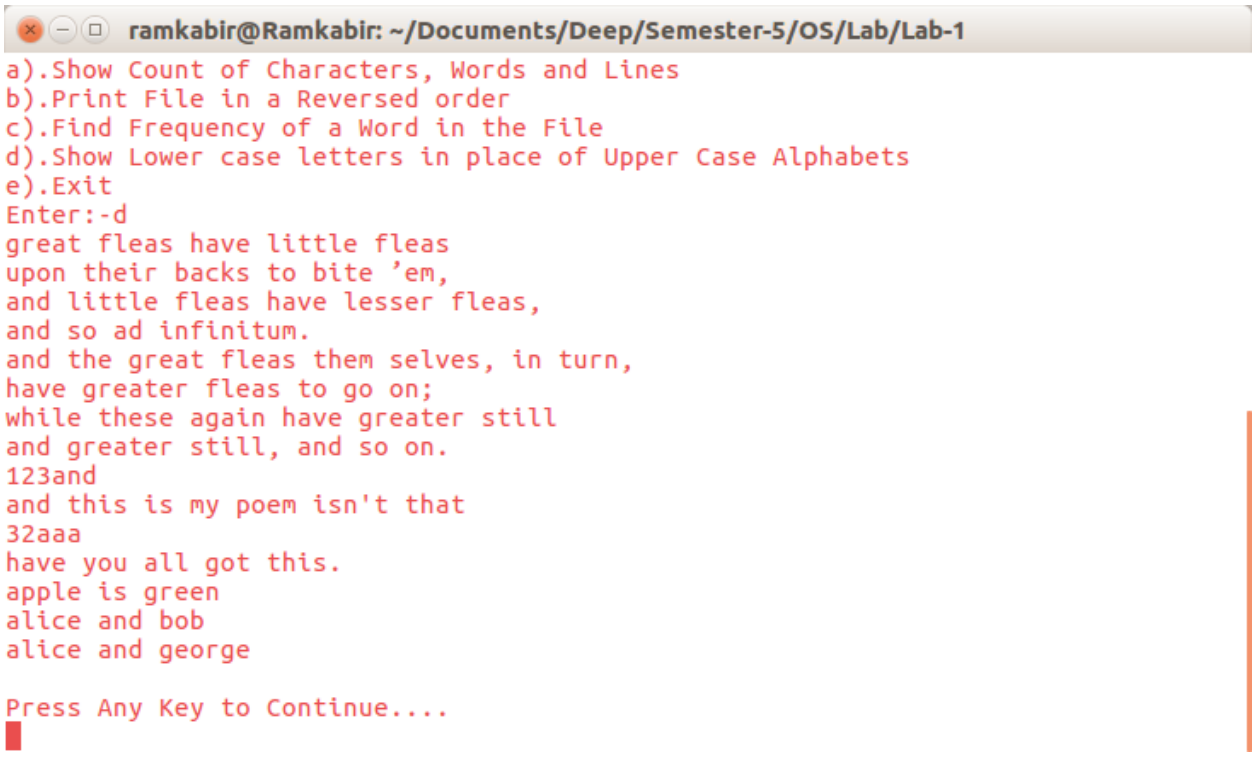
```
                count=0

                wrd=$( cat $file1 | tr [:space:] '\n' | tr -d
'[:punct:]' | grep -v "^\s*$")

                for w in $wrd
                do

                    if [ $w = $word ]
                    then

                        (( count++ ))

                    fi

                done

                echo -e "\nThe Word Appeared $count times"

                echo -e "\nPress Any Key to Continue...."
                read v

                continue;;

        [dD])   tr '[A-Z]' '[a-z]' < $file1

                echo -e "\nPress Any Key to Continue...."
                read v

                continue;;


        [eE])   echo "Bye............"
                exit;;

        *)  echo "Choice Not Found, try again..."
            continue;;

    esac

done
```

Screenshot:-


ramkabir@Ramkabir: ~/Documents/Deep/Semester-5/OS/Lab/Lab-1
Welcome, Enter From Following Options:
a).Show Count of Characters, Words and Lines
b).Print File in a Reversed order
c).Find Frequency of a Word in the File
d).Show Lower case letters in place of Upper Case Alphabets
e).Exit
Enter:-a
Lines:15
Words:70
Characters:368
Press Any Key to Continue....


ramkabir@Ramkabir: ~/Documents/Deep/Semester-5/OS/Lab/Lab-1
a).Show Count of Characters, Words and Lines
b).Print File in a Reversed order
c).Find Frequency of a Word in the File
d).Show Lower case letters in place of Upper Case Alphabets
e).Exit
Enter:-b
alice and george
Alice and bob
Apple is green
have you all got this.
32aaa
and this is my poem isn't that
123And
And greater still, and so on.
While these again have greater still
Have greater fleas to go on;
And the great fleas them selves, in turn,
And so ad infinitum.
And little fleas have lesser fleas,
Upon their backs to bite 'em,
Great fleas have little fleas
Press Any Key to Continue....


ramkabir@Ramkabir: ~/Documents/Deep/Semester-5/OS/Lab/Lab-1
Welcome, Enter From Following Options:
a).Show Count of Characters, Words and Lines
b).Print File in a Reversed order
c).Find Frequency of a Word in the File
d).Show Lower case letters in place of Upper Case Alphabets
e).Exit
Enter:-c
Enter Word:
alice
The Word Appeared 1 times
Press Any Key to Continue....


ramkabir@Ramkabir: ~/Documents/Deep/Semester-5/OS/Lab/Lab-1
a).Show Count of Characters, Words and Lines
b).Print File in a Reversed order
c).Find Frequency of a Word in the File
d).Show Lower case letters in place of Upper Case Alphabets
e).Exit
Enter:-d
great fleas have little fleas
upon their backs to bite 'em,
and little fleas have lesser fleas,
and so ad infinitum.
and the great fleas them selves, in turn,
have greater fleas to go on;
while these again have greater still
and greater still, and so on.
123and
and this is my poem isn't that
32aaa
have you all got this.
apple is green
alice and bob
alice and george
Press Any Key to Continue....

11).Script-16

Write a shell script to ask for the name of a user, and check whether that user is currently online or not

Code:-

```
#!/bin/bash

#Created by Deep C. Patel-1401010

clear

echo "Enter the Name of the User:"
read name

parsed=$( who | tr -c [:alpha:] "\n" | sort | uniq )

for nm in $parsed
do

    if [ $nm = $name ]
    then

        echo -e "\nUser Found"
        exit;

    fi

done

echo -e "\nUser Not Found"
```

Screenshot:-

---

12).Script-18

Explain output for the following

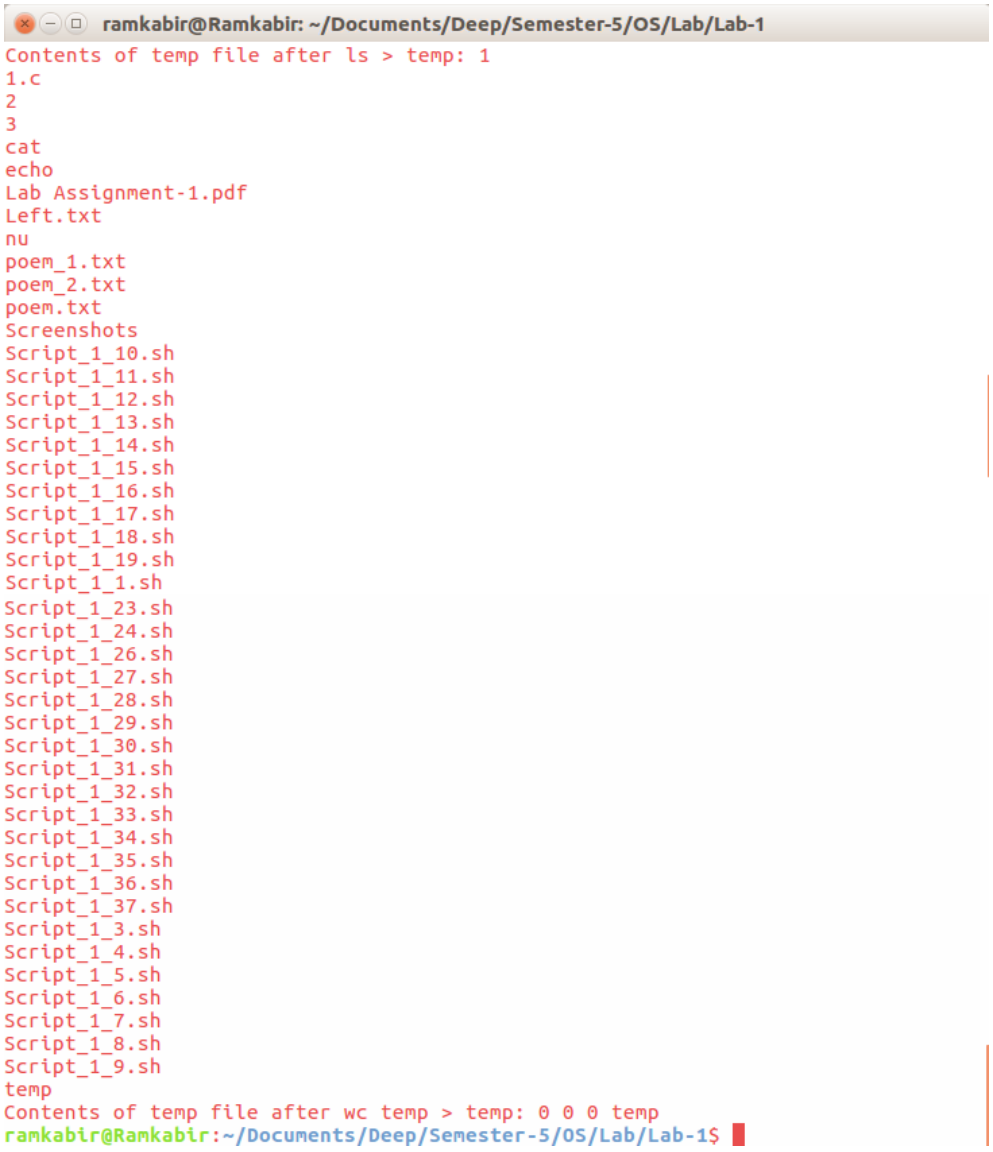
$ls > temp

$wc temp > temp

Code:-

```
#!/bin/bash

#Created by Deep C. Patel-1401010

ls > temp

echo "Contents of temp file after ls > temp: `cat temp`"
---------------------------------------------------------------
wc temp > temp
echo "Contents of temp file after wc temp > temp: `cat temp`
```

Screenshot:-



Explanation:-

1).Command creates temp file if it not exists and copies output of `$ls` command to it.

2).This command overwrites output of "`$wc temp`" in temp file. The Output consists of no of lines in temp, no of words in temp, no of characters in temp file.

---

13).Script-20

Count the users

Code:-

```
#!/bin/bash

#Created by Deep C. Patel-1401010

clear
echo -e "No of Users:\c"
who | wc -l
```

Screenshot:-

14).Script-21

Count the total files

Code:-

```
#!/bin/bash

#Created by Deep C. Patel-1401010

clear
echo -e "No of Files:\c"
expr `ls -la | wc -l` - 1
```

Screenshot:-

Note:- Here we count the files of the current directory, as no default directory was specified in the question.

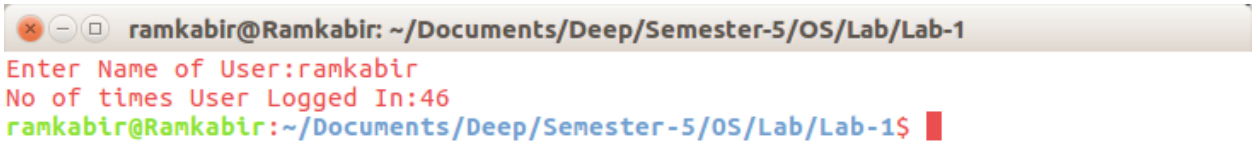
15).Script-23

Count how many times you've logged in

Code:-

```
#!/bin/bash

#Created by Deep C. Patel-1401010

clear
echo -e "Enter Name of User:\c"
read name
echo -e "No of times User Logged In:\c"
expr `last $name | wc -l` - 2
```

Screenshot:-



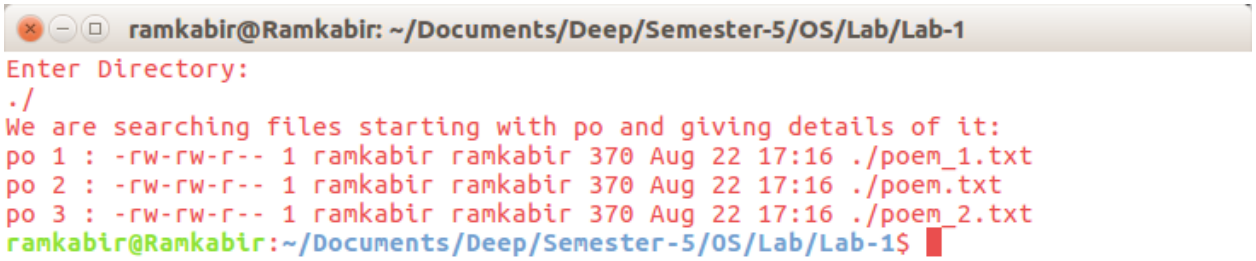
16).Script-26

List detailed attributes of all files that have names beginning with “po” followed by either 1,2,3,4, or 5

Code:-

```
#!/bin/bash

#Created by Deep C. Patel-1401010

clear
echo "Enter Directory:"
read path

echo "We are searching files starting with po and giving details
of it:"

count=1

path1=$( echo $path | find -type f )

for name in $path1
do

    name1=$( basename "$name" )
    var=$( echo $name1 | cut -c1-2 )

    if [ $var = "po" ]
    then

        echo "po $count : `ls -l $name`"

        count=$((count+1))

    fi

done
```

Screenshot:-



---

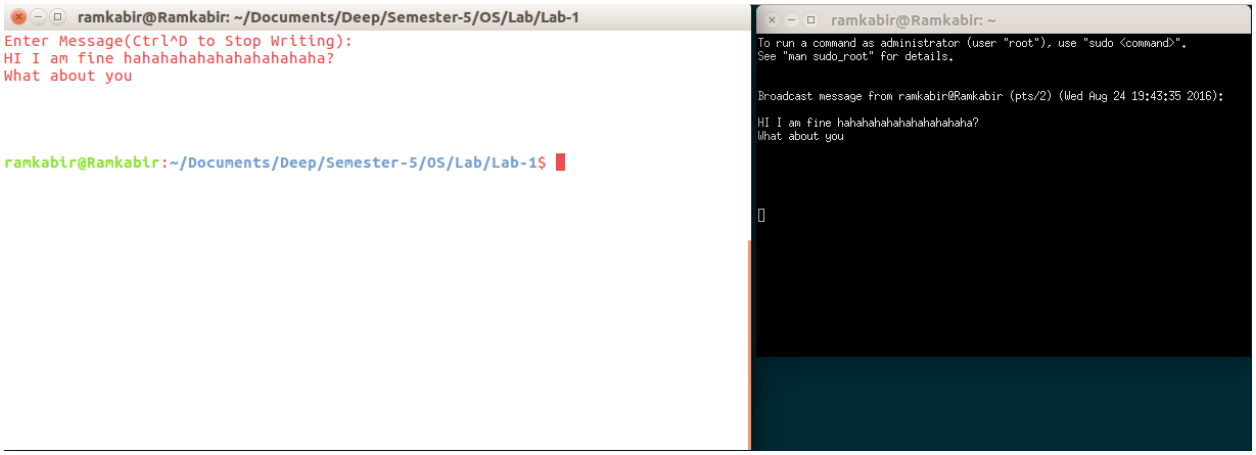
17).Script-30

Write a shell program that sends a note to several people on your system

Code:-

```
#!/bin/bash

#Created by Deep C. Patel-1401010

clear

echo "Enter Message(Ctrl^D to Stop Writing):"
wall
```

Screenshot:-



---

18).Script-32

Locate variable in C source files(i.e. with .c or .h extensions)

Code:-

```
#!/bin/bash

#Created by Deep C. Patel-1401010

clear

echo "Enter the .c/.h file Path:"
read path

echo "Enter Variable:"
read cVar

count=0

while read Line
do
    count=$((count+1))

    parsed=$( echo $Line | tr -c [:alpha:] "\n" | sort | uniq )
```

```
    for nm in $parsed
    do

        if [ $nm = $cVar ]
        then

            echo "$count This Line has your Variable"
            break
        fi

    done

done < $path
```

Snapshot:-

19).Script-35

List users without passwords

Code:-

```
#!/bin/bash

#Created by Deep C. Patel-1401010

clear

path=$( cut -d : -f1 "/etc/passwd" | sort )

for nm in $path
do

    set -- `sudo passwd -S $nm`

    if [ ! `echo $2` = "P" ]
    then
        echo $nm
    fi

done
```

Snapshot:-