4. Shell Script

a) Write a shell script that takes a command –line argument and reports on whether it is directory, a file or something else

Aim: to a shell script that takes a command –line argument and reports on whether it is directory, a file or something else

Program:

[20A91A0568@Linux ~] $ vi program.sh

```
echo "Enter a file name:"
read file
if [ -f $ file ]
then
echo "  yes  it is a File"
elif [ -d $file ]
then
echo "yes it is a Directory"
else
echo "name not in the list"
fi
```

OUTPUT:

[20A91A0568@linux~]$sh program.sh

Enter a file name:

Program.sh

Yes it is a Directory

b) write a shell script to find Factorial of a number

[20A91A0568@Linux ~]$ vi fact.sh

```
echo "enter a number:"
read num
i=1
counter=1
fact=1
while [ $num -ge $counter ]
 do
fact=`expr $fact \* $counter`
counter=`expr $counter + 1`
done
echo "the factorial of $num is : $fact"
```

OUTPUT:

[20A91A0568@Linux ~]$ sh fact.sh

```
enter a number:
5
the factorial of 5 is : 120
```

5. Shell Script

a) Write a shell script that determines the period for which a specified user is working on the system.

Aim: to a shell script that determines the period for which a specified user is working on the system .

[20A91A0568@Linux ~]$ vi user.sh

```
echo "enter the login of the user:"
read name
logindetails=`who|grep -w "$name"|grep "tty"`
if [$? -ne 0]
 then
echo "$name has not logged in yet" exit
fi

loginhours=`echo "$logindetails"|cut -c 26,27`
loginminutes=`echo "$logindetails"|cut -c 29-30`
 hournow=`date|cut -c 12,13`
minnow=`date|cut -c 15,16`
hour=`expr $loginhours-$hournow`
min=`expr $loginminute-$minnow`
echo "$name is working since $hour hrs $min minutes"
```

output:

[20A91A0568@Linux ~]$ sh user.sh

enter the login of the user:

20A91A0568

20A91A0568 is working since -11 hrs -07 minutes

5 b)shell script that accepts a file name ,starting and ending line numbers as arguments and display all the lines between the given lines

Aim:to a shell script that accepts a file name starting and ending line numbers as arguments and displays all the lines between the given line numbers.

[20A91A0568@Linux ~]$ vi displaylines.sh

```
echo "enter a filename:"
read file
echo "enter the starting line:"
read s
echo "enter the ending line:"
 read n
sed -n $s,$n\p $file|cat >newline
cat newline
```

OUTPUT

```
[20A91A0568@Linux ~]$ sh displaylines.sh
 enter a filename:
mss
enter the starting line:
1
enter the ending line:
4
 Hi
hello
aditya
hi
RK hi
```

6. Shell Script Write a shell script that computes the gross salary of a employee according to the following rules:

i) If basic salary is < 1500 then HRA =10% of the basic and DA =90% of the basic.

ii) If basic salary is >=1500 then HRA =Rs500 and DA=98% of the basic. The basic salary is entered interactively through the key board.

Aim: a shell script that computes the gross salary of a employee according to the following rules

[20A91A0568@Linux ~] $ vi salary.sh

```
echo "enter basic salary:"
 read bs
 if [ $bs -lt 1500 ]
then
hra=`echo $bs\*10/100|bc`
da=`echo $bs\*90/100|bc`
else
 hra=500
da=`echo $bs\*98/100|bc`
fi
gs=`echo $bs+$hra+$da|bc`
echo "DA $da"
 echo "HRA $hra"
 echo "gross salary $gs"
```

OUTPUT:

```
[20A91A0568@Linux ~]$ sh salary.sh
enter basic salary:
100
DA 90
HRA 10
gross salary 200
```

Q)GREP SCRIPT THAT ASKS FOR A WORD AND A FILE NAME AND TELLS HOW MANY LINES CONTAINS THAT FILE

[20A91A0568@Linux ~]$ vi hlines.sh

```
echo "enter a word:"
 read w
echo "enter a file name:"
read f
no1=`grep -c "$w" $f`
echo "the number of lines are :"$no1
```

OUTPUT:

[20A91A0568@Linux ~]$ shhlines.sh

```
enter a word: hi
enter a file name:
mss
the number of lines are :8
```

Q) TO FIND LENGTH OF A STRING USING SHELL SCRIPT

[20A91A0568@Linux ~] $ vi length.sh

```
 echo "enter a string:"
read string l=`echo $string|wc -c`
echo "length of string is =$l"
```

OUTPUT:

```
[20A91A0568@Linux ~]$ sh length.sh
 enter a string:
aditya
length of string is =6
```

Q)SHELL SCRIPT TO CONCATENATE TWO STRINGS

[20A91A0568@Linux ~] $ vi concatenate.sh

```
echo "enter a first string:"
read s1
echo "enter a second string:"
read s2
 s3=$s1$s2
echo "concatenated string is $s3"
```

OUTPUT:

```
[20A91A0568@Linux ~]$ sh concatenate.sh
enter a first string:
aditya enter a second
string:
engg
concatenated string is adityaengg
```

Q) Write a shell script to accept emp no, emp name, basic salary and find the DA, HRA, TA, PF, IT using the following rules

1. If basic salary>5000 then

HRA=18% OF BASICSAL

PF=13% OF BASICSAL

IT=14% OF BASICSAL

TA=10% OF BASICSAL

DA=35% OF BASICSAL

2. If basic salary<5000 then

HRA=550

PF=13% OF BASICSAL

IT=14% OF BASICSAL

TA=10% OF BASICSAL

DA=35% OF BASICSAL

[**20A91A0568@Linux ~]$ vi employe.sh**

```
echo "enter employee no:"
read empno
echo "enter employee name:"
read empname
echo "enter basic salary:"
read bs
if [ $bs -lt5000 ]
then
hra=550
da=`echo $bs\*35/100|bc`
pf=`echo $bs\*13/100|bc`
it=`echo $bs\*14/100|bc`
ta=`echo $bs\*10/100|bc`
else
hra=`echo $bs\*18/100|bc`
da=`echo $bs\*35/100|bc`
```

```
pf=`echo $bs\*13/100|bc`
it=`echo $bs\*14/100|bc`
ta=`echo $bs\*10/100|bc`
fi
gs=`echo $bs+$hra+$da+$pf+$it+$ta|bc`
echo "DA $da"
echo "HRA $hra"
echo "PF $pf"
echo "IT $it"
echo "TA $ta"
echo "GROSS SALARY $gs"
```

OUTPUT:

```
[20A91A0568@Linux ~]$sh employe.sh
enter employee no: 123
 enter employee name: aditya
enter basic salary:
15000
DA 5250
HRA 2700
PF 1950
IT 2100
TA 1500
GROSS SALARY 28500
[20A91A0568@Linux ~]$sh employe.sh
enter employee no: 456
enter employee name: RK
 enter basic salary:
1200
DA 420
HRA 550
PF 156
IT 168
TA 120   GROSS SALARY 2614
```

7. Shell Script

a) Write a shell script that accepts two integers as its arguments and computes the value of first number raised to the power of the second number.

Aim:to a shell script that accepts two integers as its arguments and computes the value of first number raised to the power of the second number**.**

[20A91A0568@Linux ~]$ vi power.sh

```
if [ $# -ne 2 ]
then
echo "invalid number of arguments"
exit
fi
pwr=`echo $1^$2|bc`
echo "$1 raised to $2 is $pwr"
```

OUTPUT:

```
[20A91A0568@Linux ~]$sh power.sh 2 3
2 raised to 3 is 8
```

7 b) Write a shell script which will display Armstrong number from given arguments.

Aim: to ashell script which will display Armstrong number from given arguments.

[20A91A0568@Linux ~]$ vi armstrong.sh

```
for n in $*
Do
 t=$n
sum=0
while [ $n -ne 0 ]
do
r=`expr $n % 10`
sum=`expr $sum + $r \* $r \* $r`
n=`expr $n / 10`
done
if [ $t -eq $sum ]
then
echo $t is armstrong number
else
echo $t is not armstrong number fi
done
```

OUTPUT:

```
[20A91A0568@Linux ~]$sh armstrong.sh 153
153 is armstrong number
[20A91A0568@Linux ~]$sh armstrong.sh 125
125 is not armstrong number
```

8.Shell Script

Write an interactive file-handling shell program. Let it offer the user the choice of copying, removing, renaming, or linking files. Once the user has made a choice, have the program ask the user for the necessary information, such as the file name, new name and so on.

[20A91A0568@Linux ~]$ vi filehandling.sh

```
echo 1.copy
echo 2.rename
echo 3.remove
echo 4.link
echo 5.exit
echo "enter your choice"
read ch
case $ch
in
1) echo "enter the source file"
   read s
echo "enter the destination file"
read d
cp $s $d
;;
2) echo "enter old file name"
   read of
echo "enter the new filename"
read nf mv $of $nf
;;
3) echo "enter the filename to
   delete" read df rm $df
;;
4) echo "enter file 1" read f1
   echo "enter file 2" read f2 ln
   $f1 $f2
;;
```

```
5) exit 0
;;
esac
```

OUTPUT

```
[20A91A0568@Linux ~]$sh filehandling.sh
1.copy
2.rename
3.remove
4.link 5.exit enter
your choice 1 enter
the source file
a.txt enter the
destination file
b.txt
[20A91A0568@Linux ~]$sh filehandling.sh
1.copy
2.rename
3.remove
4.link 5.exit enter
your choice 2
enter old file name
b.txt enter the new
filename
d.txt
```

2.EXPERIMENT

a) Use the cat command to create a file containing the following data.Call it mytable use tabs to separatethe fields.

1425 Ravi 15.65

4160 Ramu 26.27

6830 Sita 36.15

1450 Raju 21.86

b) Study of vi editor

c) Use the cat command to display the file, my table.

d) Use the vi command to correct any errors in the file, my table.

e) Use the sort command to sort the file my table according to the first field. Call the sorted file my table (same name).

f) Print the file my table.

a)create a table using cat command

[20A91A0568@Linux ~]$ cat>mytable

1425 Ravi 15.65

4160 Ramu 26.27

6830 Sita 36.15

1450 Raju 21.86

2.b)Study of vi editor

Aim: To Study of vi editor

vi is generally considered the de facto standard in Unix editors because −

- It's usually available on all the flavours of unix system.
- Its implementations are very similar across the board.
- It requires very few resources.
- It is more user-friendly than other editors such as the **ed** or the **ex**.

You can use the **vi** editor to edit an existing file or to create a new file from scratch. You can also use this editor to just read a text file.

Syntax: vi filename

.vi editor has three modes

1)command mode

2)insert mode

3)exit mode

1)Command mode:

Once a file is open you are in the command mode .From command mode you can:

- Invoke insert mode
- Issue editing commands
- Move cursor to a different position in the file
- Save and exit the current version of file

2)Insert mode:

In insert mode you can enter new text in the file press esc key to exit insert mode and return to command mode.

The following commands invoke the insert mode:

- a Append after cursor

A Append at the end of line

- i Insert before cursor
- l Insert at beginning of line
- r Replace character under cursor
- Open a newline above current line

3)Lastline mode:

The last vi mode is known as vi last line mode. The following command invoke exit mode.

- :q to quit (short for quit)
- :q! to quit without saving
- :wq to write and quit
- :wq! To write and quit even if file has only read permission
- X to read and quit
- :qa to quit all (short for :quit all)

Example:[20A91A05532linux~]vi factorial.sh

c) display the table using cat command

[20A91A0568@Linux ~] $ cat mytable

1425 Ravi 15.65

4160 Ramu 26.27

6830 Sita 36.15

1450 Raju 21.86

ROLL NO -20A91A0568

d)use vi command to edit

[20A91A0568@Linux kmss]$ vi mytable

e)use sort command to sort

[20A91A0568@Linux ~]$ sort -f +0 -1 mytable>new.txt

[20A91A0568@Linux ~]$ cat new.txt

1425 Ravi 15.65

1450 Raju 21.86

4160 Ramu 26.27

6830 Sita 36.15

f) print file my table

[20A91A0568@Linux ~]$ cat mytable

1425 Ravi 15.65

4160 Ramu 26.27

6830 Sita 36.15

1450 Raju 21.86

3.EXPERIMENT

a) use the appropriate command to determine your login shell.

b) use the who command and redirect result to the file called myfile1,use the more command to see the content of myfile1.

c) use the date and who command in sequence such that the output of date command will display on the screen and the output of who command is redirected to a file called myfile 2.

d)use the more command to check the content of myfile2.

a) to determine login shell

[20A91A0568@Linux ~]$ echo $SHELL

/bin/bash

b)who command and more command redirect to my file 1

[20A91A0568@Linux ~]$ who >myfile1

[20A91A0568@Linux ~]$ cat myfile1

20A91A0533 pts/0 2021-10-27 09:42 (172-7-139-250.lightspeed.irvnca.sbcglobal.net)

20A91A0534 pts/1 2021-10-27 09:42 (172-7-139-250.lightspeed.irvnca.sbcglobal.net)

20A91A0510 pts/2 2021-10-27 09:55 (172-7-139-250.lightspeed.irvnca.sbcglobal.net)

exam41 pts/3 2021-10-27 09:53 (172-7-139-250.lightspeed.irvnca.sbcglobal.net)

[20A91A0568@Linux ~]$ more myfile1

20A91A0533 pts/0 2021-10-27 09:42 (172-7-139-250.lightspeed.irvnca.sbcglobal.net)

20A91A0534 pts/1 2021-10-27 09:42 (172-7-139-250.lightspeed.irvnca.sbcglobal.net)

20A91A0510 pts/2 2021-10-27 09:55 (172-7-139-250.lightspeed.irvnca.sbcglobal.net)

exam41 pts/3 2021-10-27 09:53 (172-7-139-250.lightspeed.irvnca.sbcglobal.net)

c)date and who command on same file

[20A91A0568@Linux ~] $ date; who >myfile2

Wed Oct 27 11:20:57 IST 2021

[20A91A0568@Linux ~] $ cat myfile2

20A91A0533 pts/0 2021-10-27 09:42 (172-7-139-250.lightspeed.irvnca.sbcglobal.net)

20A91A0534 pts/1 2021-10-27 09:42 (172-7-139-250.lightspeed.irvnca.sbcglobal.net)

20A91A0510 pts/2 2021-10-27 09:55 (172-7-139-250.lightspeed.irvnca.sbcglobal.net)

d)more command to check content in myfile 2

[20A91A0568@Linux ~] $ more myfile2

20A91A0533 pts/0 2021-10-27 09:42 (172-7-139-250.lightspeed.irvnca.sbcglobal.net)

20A91A0534 pts/1 2021-10-27 09:42 (172-7-139-250.lightspeed.irvnca.sbcglobal.net)

20A91A0510 pts/2 2021-10-27 09:55 (172-7-139-250.lightspeed.irvnca.sbcglobal.net)

exam41 pts/3 2021-10-27 09:53 (172-7-139-250.lightspeed.irvnca.sbcglobal.net)