

The image shows a terminal window with the title bar "junk.sh". The window contains a shell script named "program 1". The script starts with "clear" and a loop "for i in \*". Inside the loop, it checks if a file is a directory using "[ -f \$i ]". If it is, it runs a command to list files and cut the fifth field: "ls -l \$i | cut -d " " -f5". It then checks if the result is equal to 0 using "[ \$s -eq 0 ]". If true, it prints "\$i is Junk File" and removes the file with "rm -i \$i". The script ends with two "fi"s and a "done".

```
clear
for i in *
do
if [ -f $i ]
then
s='ls -l $i | cut -d " " -f5'
if [ $s -eq 0 ]
then
echo "$i is Junk File"
rm -i $i
fi
fi
done
```



Applications Places System



Thu Apr 4, 12:48 PM



abcd.sh (~/Desktop/ankita) - gedit

File Edit View Search Tools Documents Help



Open ▾



Save



Undo



junk.sh X

abcd.sh X

empty file add one



Loading file '/home/rpimrd/Desktop/ankita/abcd.sh'...

sh ▾ Tab Width: 8 ▾ Ln 1, Col 1

rpimrd

Desktop

ankita

abcd.sh (~/D...

rpimrd@loc...

A screenshot of a Linux desktop environment. At the top, there is a menu bar with icons for Applications, Places, System, and several system status indicators. Below the menu bar is a terminal window titled "abcd.sh (~/Desktop/ankita) -ged". The terminal window has a dark background and contains the following text:

```
abcd.sh is Junk File
rm: remove regular empty file `abcd.sh'? y
[rcpimrd@localhost ankita]$ abcd.sh
bash: abcd.sh: command not found
[rcpimrd@localhost ankita]$ sh abcd.sh
sh: abcd.sh: No such file or directory
[rcpimrd@localhost ankita]$
```

At the bottom of the terminal window, there is a large watermark-like text that reads "1st Program - Output".

File Edit View Search Tools Documents Help



Save



Undo



arithmetic1.sh

## Program 2

```
echo enter 2 no
read a b
i=1
while true;
do
echo "1.Addition"
echo "2.Substraction"
echo "3.multiplication"
echo "4.division"
echo "5.exit"
echo enter choice
read ch
case $ch in
1)c=$(expr $a + $b)
echo add is $c
;;
2)c=$(expr $a - $b)
echo sub is $c
;;
3)c=$(expr $a '*' $b)
echo mul is $c
;;
4)c=$(expr $a / $b)
echo div is $c
```

File Edit View Search Tools Documents Help



Open



Save



Undo



arithmetic1.sh X

```
done x.addition
echo "2.Subtraction"
echo "3.multiplication"
echo "4.division"
echo "5.exit"
echo enter choice
read ch
case $ch in
1)c=$(expr $a + $b)
echo add is $c
;;
2)c=$(expr $a - $b)
echo sub is $c
;;
3)c=$(expr $a '*' $b)
echo mul is $c
;;
4)c=$(expr $a / $b)
echo div is $c
;;
5)echo "exit"
exit 0;
esac
done
```

rcpimr

rcpimrd@localhost:~

File Edit View Search Terminal Help

[rcpimrd@localhost ~]\$ sh arithmetic1.sh

enter 2 no

12 34

1.Addition

2.Substraction

3.multiplication

4.division

5.exit

enter choice

1

add is 46

1.Addition

2.Substraction

3.multiplication

4.division

5.exit

enter choice

2

sub is -22

1.Addition

2.Substraction

3.multiplication

4.division

5.exit

file.sh

five.sh

add -5

# Output

Applications Places System

n1.sh (~/D)

File Edit View Search Tools Documents Help

Open Save Undo

n1.sh X

# Program 3

```
echo enter the number
read n
rev=0
temp=$n
while [ $temp -gt 0 ]
do
rem=`expr $temp % 10`
rev=`expr $rev '*' 10 + $rem`
temp=`expr $temp / 10`
done
if [ $n -eq $rev ]
then
echo number is palindrome
else
echo number is not palindrome
fi
```

[RCPIMRD@localhost:...]

n1.sh (~/Desktop) - ge...



```
RCPIMRD@localhost:~/Desktop
File Edit View Search Terminal Help
[RCPIMRD@localhost Desktop]$ sh n1.sh
enter the number
453
number is not palindrom
[RCPIMRD@localhost Desktop]$
```

# Output

supports mouse pointer integration. This means that you do not need to capture the mouse pointer to be able to use it in your guest OS -- all mouse actions you perform will

Applications Places System

\*mcq2.sh (~/dhanshri 39) -ged

File Edit View Search Tools Documents Help

Open Save Undo

\*mcq2.sh

```
clear
echo Questions:
sc=0
echo "1 : Which is the Capital of India?"
echo "Options : a)Delhi b)Mumbai c)Nagpur d)Dhule"
read key
if test $key = "a"
then
echo "Your Answer Is Correct"
sc=`expr $sc + 10`
else
echo "Your Answer Is Incorrect"
fi
echo "2 : Which is the largest river in world?"
echo "Options : a)Ganga b)Yamuna c)Nile d>Panza"
read key
if test $key = "c"
then
echo "your answer is correct"
sc=`expr $sc + 10`
else
echo "your answer is incorrect"
fi
echo "#2 - how many keywords in C language?"
```

\*mcq2.sh (~/dhanshri 39) -ged

rcpimrd dhanshri 39 \*mcq2.sh (~/dhan...

# Program 4



```
echo "your answer is correct"
sc=`expr $sc + 10`
else
echo "your answer is incorrect"
fi
echo "3 : how many keywords in C language?"
echo "Options : a)40 b)32 c)33 d)34"
read key
if test $key = "b"
then
echo " your answer is correct "
sc=`expr $sc + 10`
else
echo " your answer is incorrect"
fi
if [ $sc -gt 0 ]
then
echo "congratulation"
echo "your score : $sc"
else
echo "Sorry"
echo "your score : $sc"
fi
```

dhanshri 39 sh Tab Width: 80

rcpimrd dhanshri 39 \*mcq2.sh (~/dhan...)

```
rcpimrd@localhost:~/dhanshri 39
File Edit View Search Terminal Help
Questions:
1 : Which is the Capital of India?
Options : a)Delhi b)Mumbai c)Nagpur d)Dhule
a
Your Answer Is Correct
2 : Which is the largest river in world?
Options : a)Ganga b)Yamuna c)Nile d)Panzra
a
your answer is incorrect
3 : how many keywords in C language?
Options : a)40 b)32 c)33 d)34
b
    your answer is correct
congratulation
your score : 20
[rcpimrd@localhost dhanshri 39]$
```

## Output

RedhatLinux [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

You have the Auto capture keyboard option turned on. This will cause the Virtual Machine to automatically capture the keyboard every time this VM window is activated and make it unavailable to other applications running on your host machine when this VM window has focus.

Application Places Home

dos.sh (~) - gedit

File Edit View Search Tools Documents Help

Open Save Undo Cut Copy Paste

dos.sh x Program 5

```
echo "dir"
echo "date"
echo "cls"
echo "md"
echo "exit"
while [ 1 ]
do
echo -e "C:>"
read n
case $n in
dir) ls ;;
date)date ;;
del) echo -e "\n\n Enter the file name you which want to delete "
read fn
rm -i $fn
ls ;;
cls)clear ;;
md)echo -e "\n\n give new directory name "
read d
mkdir $d
ls ;;
exit)exit ;;
*)echo entered wrong command
esac
```

34°C  
Smoke



dos.sh

```
date
echo "cls"
echo "md"
echo "exit"
while [ 1 ]
do
echo -e "C:\>"
read n
case $n in
dir) ls ;;
date)date ;;
del) echo -e "\n\n Enter the file name you which want to delete "
read fn
rm -i $fn
ls ;;
cls)clear ;;
md)echo -e "\n\n give new directory name "
read d
mkdir $d
ls ;;
exit)exit ;;
*)echo entered wrong command
esac
done
```

I



ture keyboard option turned on. This will cause the Virtual Machine to automatically capture the keyboard every time the VM window is activated and make it unavailable to other applications running on your host machine; when the keyboard is captured by the VM, it is not available to other applications.

Applications Places System

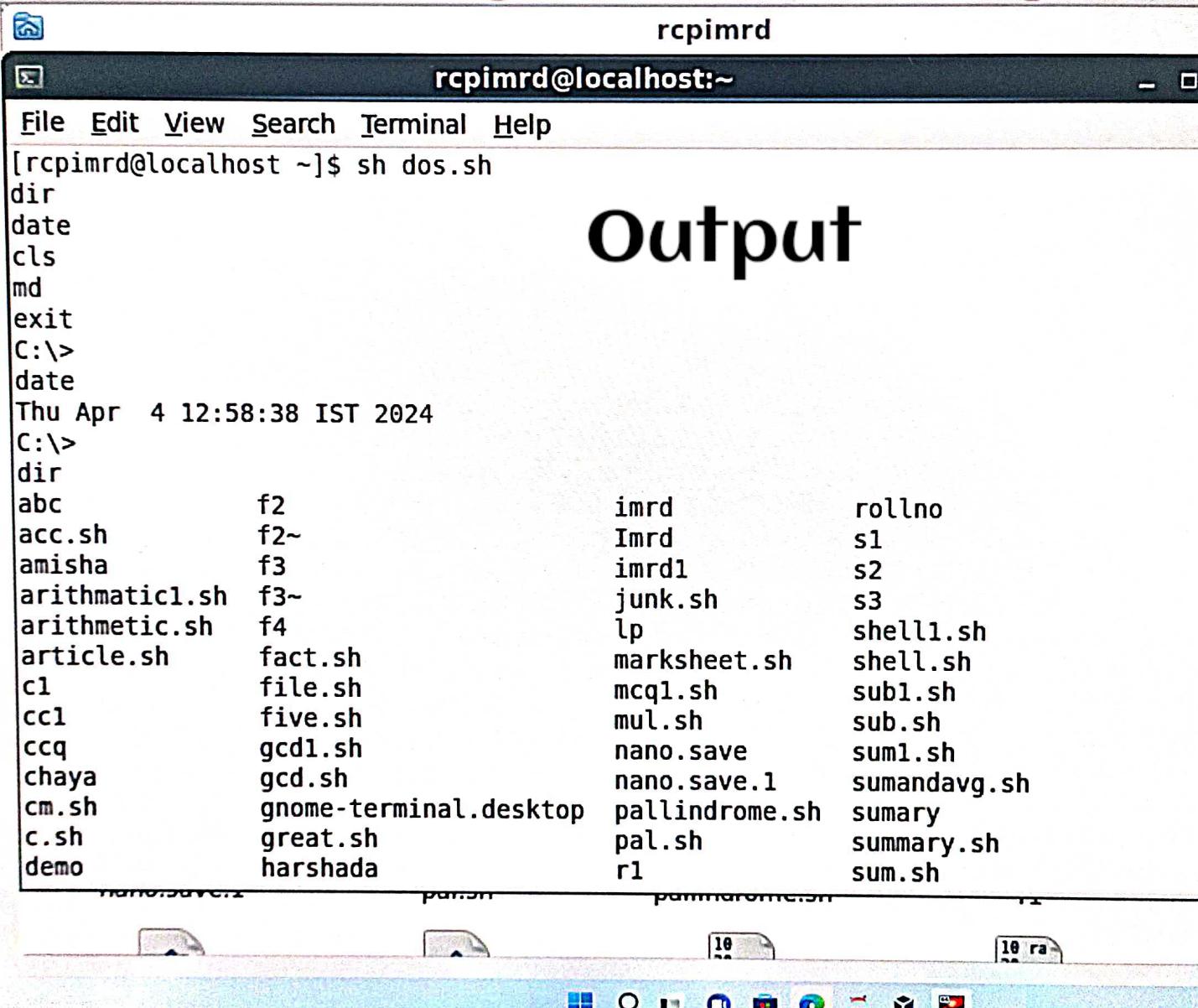
rcpimrd

rcpimrd@localhost:~

File Edit View Search Terminal Help

```
[rcpimrd@localhost ~]$ sh dos.sh
dir
date
cls
md
exit
C:\>
date
Thu Apr  4 12:58:38 IST 2024
C:\>
dir
abc          f2                  imrd      rollno
acc.sh       f2~                 Imrd      s1
amisha       f3                  imrdl     s2
arithmatic1.sh f3~               junk.sh   s3
arithmetric.sh f4                lp        shell1.sh
article.sh    fact.sh            marksheet.sh shell.sh
c1           file.sh            mcq1.sh   sub1.sh
cc1          five.sh            mul.sh    sub.sh
ccq          gcd1.sh            nano.save sum1.sh
chaya        gcd.sh             nano.save.1 sumandavg.sh
cm.sh         gnome-terminal.desktop pallindrome.sh summary
c.sh          great.sh           pal.sh    summary.sh
demo         harshada           r1       sum.sh
```

Output



[running] - Oracle VM VirtualBox  
File View Input Devices Help

Machine reports that the guest OS supports mouse pointer integration. This means that you do not need to capture the mouse pointer to be able to use it in your guest OS -- all mouse actions you perform when the r

Applications Places System

summary.sh (~/dhanshri 39) - gedit

File Edit View Search Tools Documents Help

Open Save Undo

summary.sh

# Program 6

```
clear
echo "files with words <= 100 are" >> summary
echo
for i in *
do
if [ -f $i ]
then
words=`cat $i | wc -w`
if [ $words -le 100 ]
then
echo $i $words >> summary
fi
fi
done
echo
echo "files with words > 100 & < 500 are" >> summary
echo
for i in *
do
if [ -f $i ]
then
words=`cat $i | wc -w`
if [ $words -gt 100 -a $words -lt 500 ]
```

rcpimrd

dhanshri 39

sh Tab Width: 8



```
sumary.sh
if [ -f $i ]
then
words=`cat $i | wc -w`
if [ $words -le 100 ]
then
echo $i $words >> sumary
fi
fi
done
echo
echo "files with words > 100 & < 500 are" >> sumary
echo
for i in *
do
if [ -f $i ]
then
words=`cat $i | wc -w`
if [ $words -gt 100 -a $words -lt 500 ]
then
echo $i $words >> sumary
fi
fi
done
```

rcpmru@localhost ~

File Edit View Search Terminal Help

```
c.sh 40
div.sh 32
dos.sh 73
f1 3
f1~ 2
f2 6
f2~ 6
f3 12
f3~ 12
f4 12
fact.sh 30
file.sh 89
five.sh 33
gcd1.sh 40
gcd.sh 46
great.sh 45
junk.sh 40
mul.sh 9
nano.save 0
nano.save.1 0
pallindrome.sh 36
pal.sh 48
r1 0
ram 13
```

**Output**

run program : sh summary.sh  
enter  
cat summary  
enter

I



Windows taskbar icons: File Explorer, File Explorer, File Explorer, File Explorer, File Explorer.



```
echo enter two num
read n1
read n2
while [ $n1 -gt $n2 ]
do
if [ $n1 -gt $n2 ]
then
n1=`expr $n1 - $n2`
else
n2=`expr $n2 - $n1`
fi
done
echo gcd of given num is $n2
```

File Edit View Search Terminal Help

[RCPIMRD@localhost Desktop]\$ sh gcd.sh  
enter two num

54

65

RCPIM gcd of given num is 65

[RCPIMRD@localhost Desktop]\$ █

# Output

Applications Places System

matrix.sh (~/Desktop/va)

File Edit View Search Tools Documents Help

Open Save Undo

gcd.sh matrix.sh

```
clear
echo "enter the element of matrix"
i=0
while [ $i -lt 9 ]
do
read mtx[i]
i=`expr $i + 1`
done
echo "the given matrix is"
echo ${mtx[0]}""${mtx[1]}""${mtx[2]}
echo ${mtx[3]}""${mtx[4]}""${mtx[5]}
echo ${mtx[6]}""${mtx[7]}""${mtx[8]}
echo ""
row1=`expr ${mtx[0]} + ${mtx[1]} + ${mtx[2]}`
row2=`expr ${mtx[3]} + ${mtx[4]} + ${mtx[5]}`
row3=`expr ${mtx[6]} + ${mtx[7]} + ${mtx[8]}`
clm1=`expr ${mtx[0]} + ${mtx[3]} + ${mtx[6]}`
clm2=`expr ${mtx[1]} + ${mtx[4]} + ${mtx[7]}`
clm3=`expr ${mtx[2]} + ${mtx[5]} + ${mtx[8]}`
totalrow=`expr $row1 + $row2 + $row3`
totalclm=`expr $clm1 + $clm2 + $clm3`
echo "the addition of total row is:"
echo $totalrow
echo "the addition of total column is"
```

[RCPIMR...]

[RCPIMR...]

[RCPIMR...]

sh matrix.sh...

## Program 8

File Edit View Search Tools Documents Help



Open



Save



Undo

gcd.sh

matrix.sh

```
while [ $i -lt 9 ]
do
read mtx[i]
i=`expr $i + 1`
done
echo "the given matrix is"
echo ${mtx[0]}""${mtx[1]}""${mtx[2]}
echo ${mtx[3]}""${mtx[4]}""${mtx[5]}
echo ${mtx[6]}""${mtx[7]}""${mtx[8]}
echo ""
row1=`expr ${mtx[0]} + ${mtx[1]} + ${mtx[2]}`
row2=`expr ${mtx[3]} + ${mtx[4]} + ${mtx[5]}`
row3=`expr ${mtx[6]} + ${mtx[7]} + ${mtx[8]}`
clm1=`expr ${mtx[0]} + ${mtx[3]} + ${mtx[6]}`
clm2=`expr ${mtx[1]} + ${mtx[4]} + ${mtx[7]}`
clm3=`expr ${mtx[2]} + ${mtx[5]} + ${mtx[8]}`
totalrow=`expr $row1 + $row2 + $row3`
totalclm=`expr $clm1 + $clm2 + $clm3`
echo "the addition of total row is:"
echo $totalrow
echo "the addition of totalclm is:"
echo $totalclm
```

sh ▾ Tab Wid



[RCPIMR...]



[RCPIMR...]



[RCPIMR...]



matrix.sh...

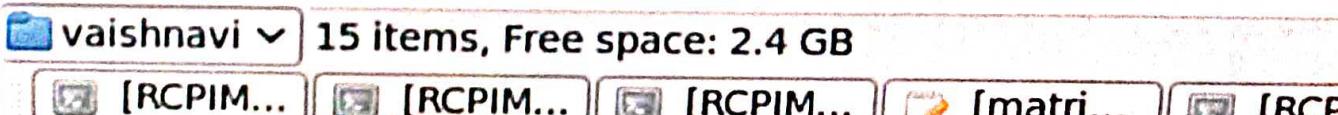


[RCPI

```
file edit view search terminal help
enter the element of matrix
3
4
5
6
7
8
9
2
3
the given matrix is
345
678
923

the addition of total row is:
47
the addition of totalclm is:
47
[RCPIMRD@localhost vaishnavi]$
```

# Output



A screenshot of a Linux desktop environment, likely Kali Linux, showing a terminal window titled "Program 9". The terminal displays a shell script named "fibo.sh" which generates a Fibonacci series. The script prompts the user for a number, initializes variables "a=0" and "b=1", and then enters a loop where it prints the value of "a", calculates the sum of "a" and "b" as "fn", updates "a" to "b", and updates "b" to "fn".

```
echo "enter the nmur:"  
read n  
a=0  
b=1  
echo "fibo series:"  
for((i=0;i<n;i++))  
do  
echo "$a"  
fn=$((a+b))  
a=$b  
b=$fn  
done
```

```
1  
1  
2  
3  
5
```

```
[RCPIMRD@localhost vaishnavi]$ sh fibo.sh  
enter the nm:u:
```

```
10
```

```
fibo series:
```

```
0
```

```
1
```

```
1
```

```
2
```

```
3
```

```
5
```

```
8
```

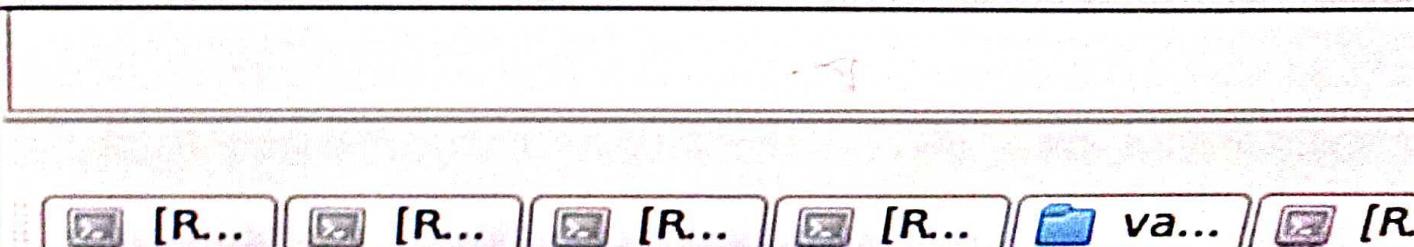
```
13
```

```
21
```

```
34
```

# Output

```
[RCPIMRD@localhost vaishnavi]$ █
```



RedhatLinux [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

You have the Auto capture keyboard option turned on. This will cause the Virtual Machine to automatically capture the keyboard every time the VM window is activated and make it unavailable to the host system.



sumandavg.sh

File Edit View Search Tools Documents Help



Open



Save



Undo

sumandavg.sh

# Program 10

```
sum=0
for i in $*
do
sum=`expr $sum + $i`
done
avg=`expr $sum / $#`
echo the total sum is $sum
echo the total average is $avg
```

34°C  
Smoke



keyboard option turned on. This will cause the Virtual Machine to automatically capture the keyboard every time the VM window is activated and make it unavailable to other applications running on your host machine. wh

The screenshot shows a terminal window titled "rcpimrd" with the command line "rcpimrd@localhost:~". The window contains the following text:

```
[rcpimrd@localhost ~]$ chmod +x sumandavg.sh
[rcpimrd@localhost ~]$ ./sumandavg.sh 10 3
the total sum is 13
the total average is 6
[rcpimrd@localhost ~]$
```

The word "Output" is overlaid in large black font across the bottom left of the terminal window.

Output

The image shows a screenshot of a Linux desktop environment. At the top, there is a menu bar with icons for Applications, Places, System, and several other applications like Evolution, Nautilus, and Gedit. Below the menu bar is a toolbar with icons for Open, Save, Undo, and Redo. The main window is titled "fact.sh (~/Desktop/va...)" and has a menu bar with File, Edit, View, Search, Tools, Documents, and Help. The window contains a text editor with the following code:

```
fact.sh x Program 11
echo "enter the num"
read n
fact=1
for((i=2;i<=n;i++))
{
    fact=$((fact*i))
}
echo $fact
```

The terminal window at the bottom has a status bar showing "sh ~".

A screenshot of a Linux desktop environment. At the top, there is a black header bar with icons for Applications, Places, System, and several system status indicators. Below the header bar is a dark grey panel containing a file icon and the user name "vaishnavi". The main area is a terminal window titled "RCPIMRD@localhost:~/Desktop/vaishnavi". The terminal window has a menu bar with File, Edit, View, Search, Terminal, and Help. The command line shows the user running a script named "fact.sh" and entering the number 5, which results in the output 120.

```
[RCPIMRD@localhost vaishnavi]$ sh fact.sh
enter the num
5
120
[RCPIMRD@localhost vaishnavi]$
```

# Output