UNIVERSITY INSTITUTE OF ENGINEERING

&

TECHNOLOGY

MAHARISHI DAYANAND UNIVERSITY , ROHTAK

PRACTICAL FILE

OPERATING SYSTEM

SUBMITTED TO :

DR. KAMNA SOLANKI

(ASSISTANT PROFESSOR CSE DEPTT.)

SUBMITTED BY:

SHUBHI, CSE-1

23534

[UIET , MDU]

INDEX

UNIX

1.HISTORY:

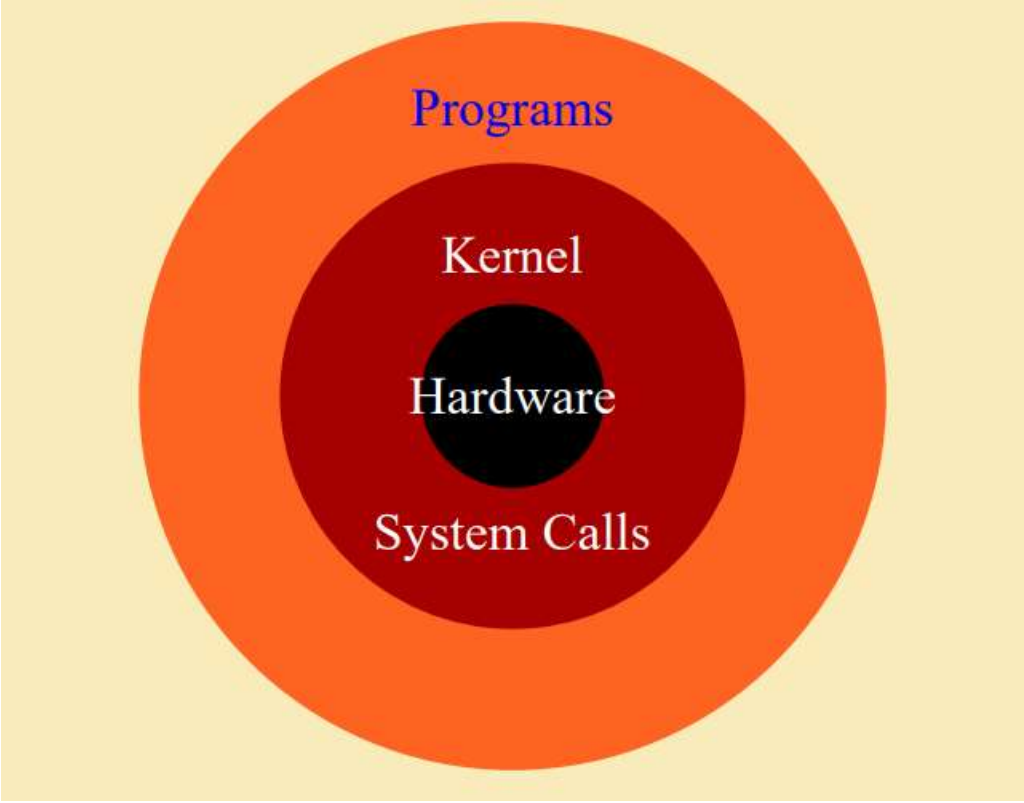
1969	multics project (MIT, GE, AT&T)
1970s	AT&T Bell Labs
1970s/1980s	UC Berkeley
1980s	DOS imitated many Unix ideas Commercial Unix fragmentation GNU Project
1990s	Linux
Now	Unix is widespread and available from many sources, both free and commercial

2.UNIX SYSTEMS:

SunOS/Solaris	**Sun Microsystems**
Digital Unix (Tru64)	Digital/Compaq
HP-UX	Hewlett Packard
Irix	SGI

UNICOS	Cray
NetBSD, FreeBSD	UC Berkeley / the Net
Linux	Linus Torvalds / the Net

3. UNIX Structure, The Operating System:



4. UNIX PROGRAMS:

• Shell is the command line interpreter

• Shell is just another program

A program or command ,

• interacts with the kernel

• may be any of:

  – built-in shell command

  – interpreted script

  – compiled object code file

Getting started:

• Login and password prompt to log in

• Login is user's unique name

• password is changeable, known only to user not to system staff

• Unix is case sensitive

• issued login and password (usually in lower case)

EXITING:

^C - interrupt

^D - can log a user off , frequently disabled logout - leave the system

exit - leave the shell

4.Some COMMANDS and their FUNCTIONS:

Commands	*Functions*
Cd	Change directory
Pwd	Print working directory
Mkdir	Make a new subdirectory
Rmdir	Remove a directory
Ls	List files in directory
Mv	Rename (move) a file
Cp	Copy a file
Rm	Delete (remove) a file
Cat	Outputs the content of a file to the screen
file	Identify the type of a file
Tail	Display the last few lines of a text file
Chmod	Change access permissions files chmod file name

Head	Display the first few lines of a text file
Ln	Creates symbol link
Passwd	

5. Permissions on file:

chmod [options] file

chmod u+w file

gives the user (owner) write permission

chmod g+r file

gives the group read permission

chmod o-x file

removes execute permission for others

List directory contents:

ls [options] [argument]

-a list all files

-d list directory itself, not contents

-l long listing (lists mode, link info, owner, size, last modification

-g unix group (requires -l option)

Permissions:

r read permission

w write permission

x execute permission

- no permission

Change permissions on file:

chmod [options] file

using numeric representations for permissions:

r = 4

w = 2

x = 1

Total: 7

chmod [options] file:

chmod 7 7 7 filename

user group others

gives user, group, and others r, w, x permissions

Display COMMANDS:

echo echo the text string to stdout

cat concatenate (list)

head display first 10 (or #) lines of file

tail display last 10 (or #) lines of file

User Listing:

who [am i]

% who	lists all users currently on system
% who am i	reports information on command user
% whoami	reports username of command user

6.SHELLS

The shell sits between you and the operating system

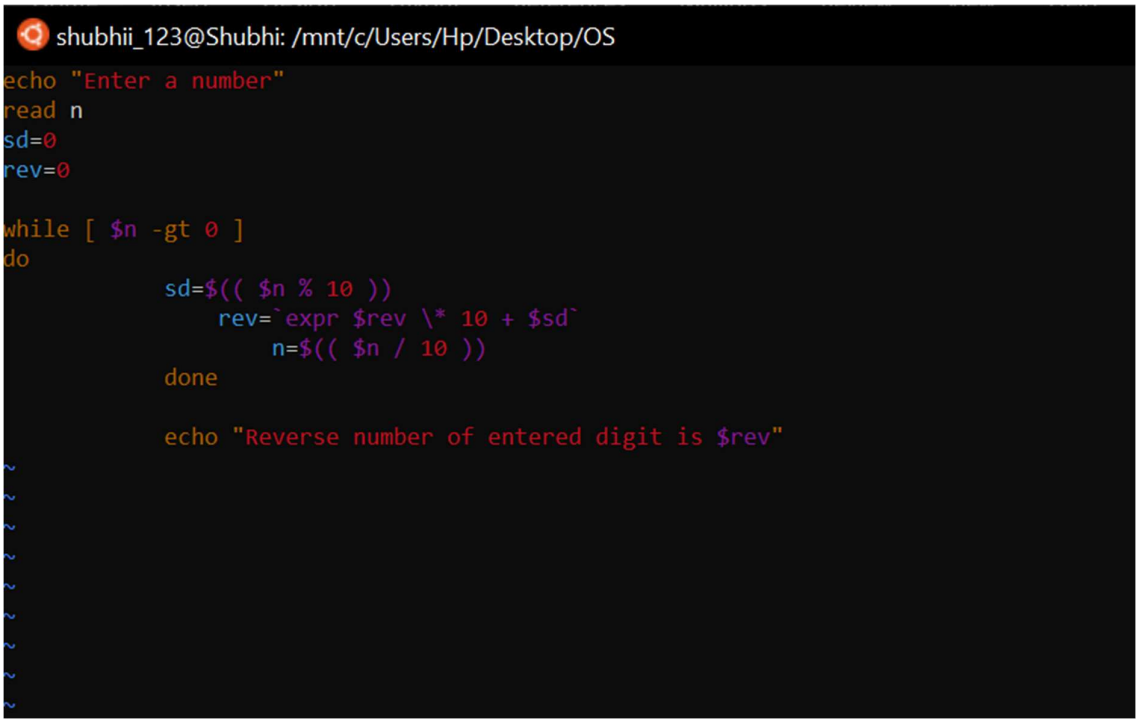
- acts as a command interpreter
- reads input
- translates commands into actions to be taken by the system

Now , we will see some programs in UBUNTU:

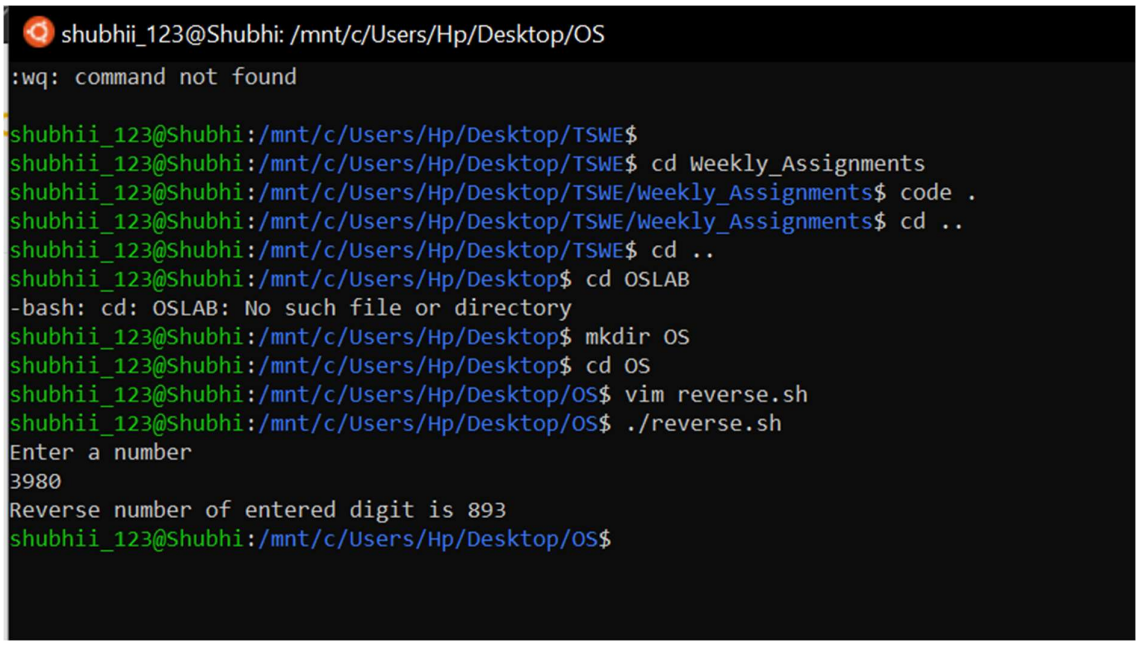
PROGRAM-1

To print the reverse of an input number:

PROGRAM CODE:



OUTPUT:


shubhii_123@Shubhi: /mnt/c/Users/Hp/Desktop/OS
:wq: command not found
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/TSWE$
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/TSWE$ cd Weekly_Assignments
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/TSWE/Weekly_Assignments$ code .
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/TSWE/Weekly_Assignments$ cd ..
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/TSWE$ cd ..
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop$ cd OSLAB
-bash: cd: OSLAB: No such file or directory
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop$ mkdir OS
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop$ cd OS
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/OS$ vim reverse.sh
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/OS$ ./reverse.sh
Enter a number
3980
Reverse number of entered digit is 893
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/OS$

PROGRAM-2

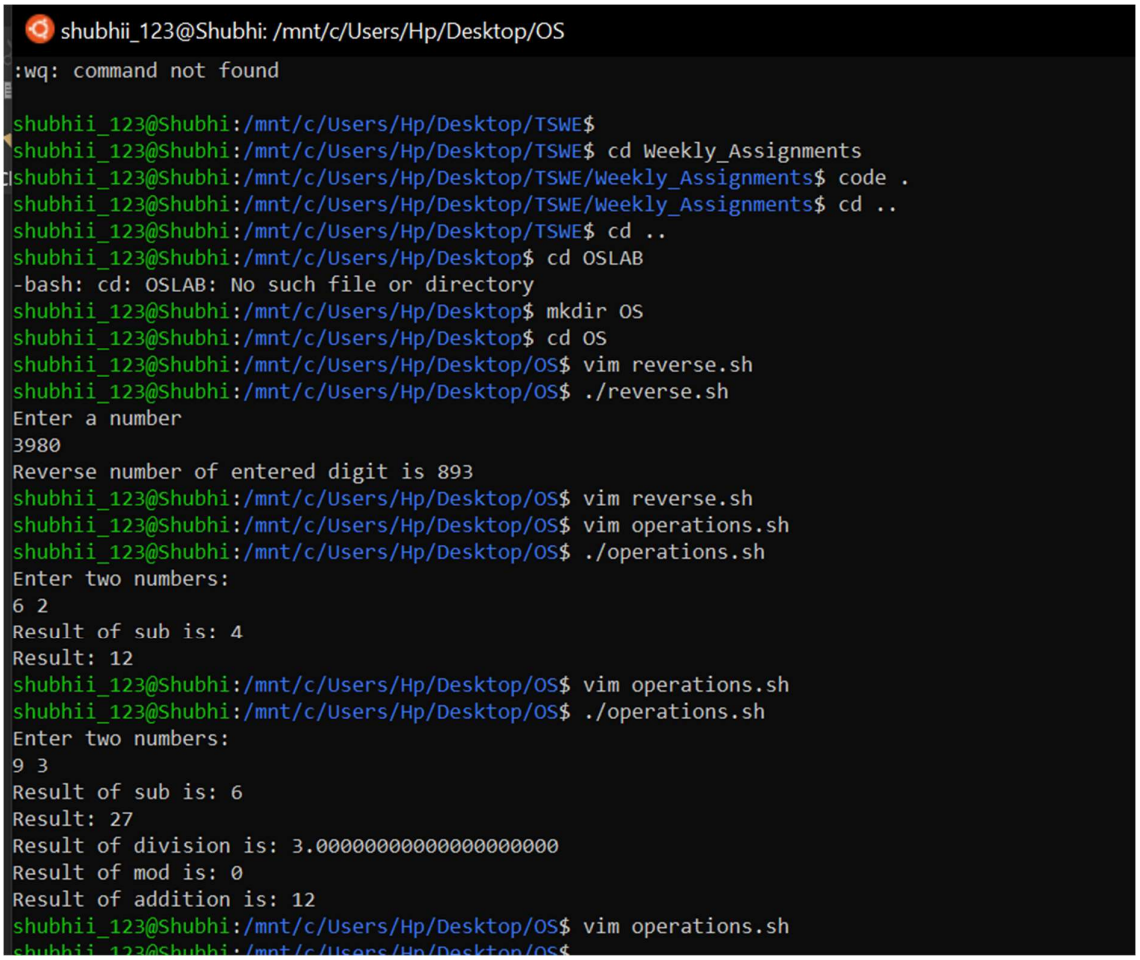
To calculate all arithematic operations on two numbers:

PROGRAM CODE:

shubhii_123@Shubhi: /mnt/c/Users/Hp/Desktop/OS

```
#Enter two numbers from user
echo "Enter two numbers: "
read a b

# compute subtraction result
result=`expr "$a - $b" | bc`
echo "Result of sub is: $result"

result=`expr "$a * $b" | bc`
# print output
echo "Result: $result"

result=`expr "$a / $b" | bc -l`

# print output
echo "Result of division is: $result"


# compute modulus result
result=`expr "$a % $b" | bc`

# print output
echo "Result of mod is: $result"


# perform addition
result=`expr "$a + $b" | bc`

# show result
echo "Result of addition is: $result"
```

OUTPUT:



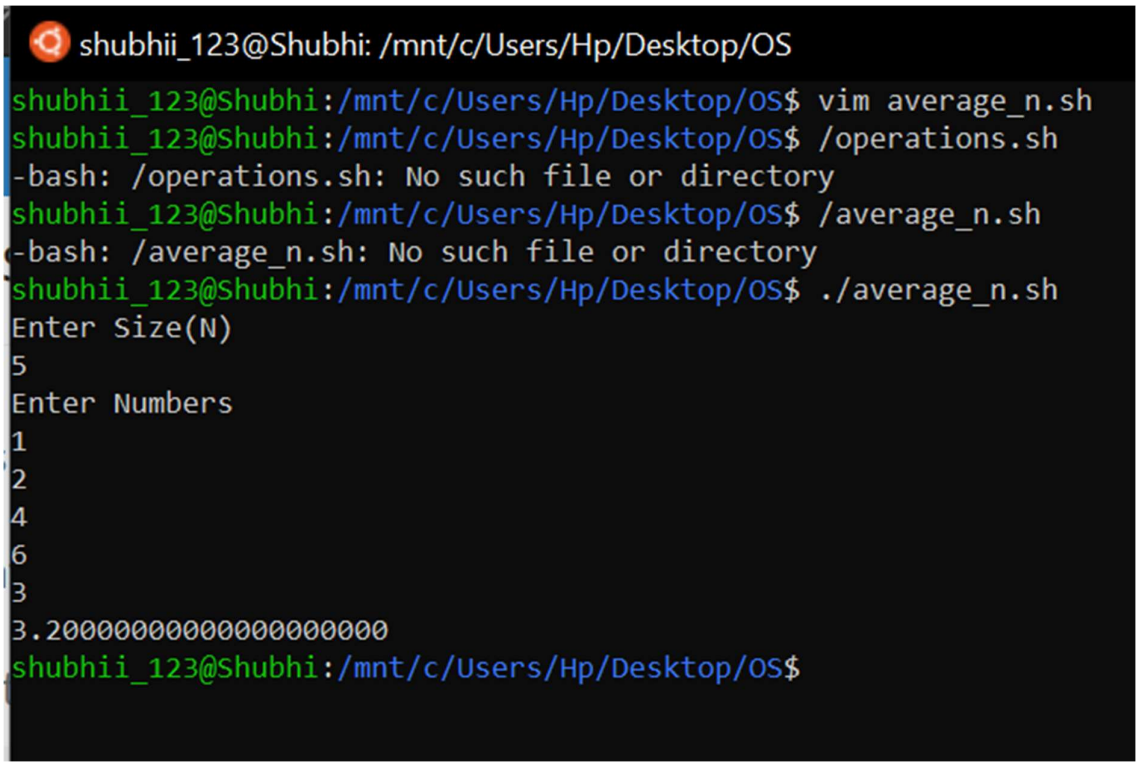
PROGRAM-3

To find average of N numbers entered by user

PROGRAM:


shubhii_123@Shubhi: /mnt/c/Users/Hp/Desktop/OS
echo "Enter Size(N)"
read N
i=1
sum=0
echo "Enter Numbers"
while [$i -le $N]
do
read num #get number
sum=$((sum + num)) #sum+=num
i=$((i + 1))
done
avg=$(echo $sum / $N | bc -l)
echo $avg

OUTPUT:



PROGRAM-4

To calculate percentage and grade of the student

PROGRAM :

shubhii_123@Shubhi: /mnt/c/Users/Hp/Desktop/OS

```
echo "Enter the five subject marks for the student"
read m1 m2 m3 m4 m5
sum1=`expr $m1 + $m2 + $m3 + $m4 + $m5`
echo "Sum of 5 subjects are: " $sum1
per=`expr $sum1 / 5`
echo " Percentage: " $per
if [ $per -ge 60 ]
then
        echo "You get Distinction”
elif [ $per -ge 50 ]
then
        echo “You get First class”
elif [ $per -ge  40 ]
then
        echo "You get Second class"
else
             echo "You get Fail"
fi
~
~
```

OUTPUT:

```
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/OS$ vim marksheet.sh
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/OS$ ./marksheet.sh
Enter the five subject marks for the student
80 90 80 70 60
Sum of 5 subjects are:  380
 Percentage:  76
You get Distinction
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/OS$
```

PROGRAM-5

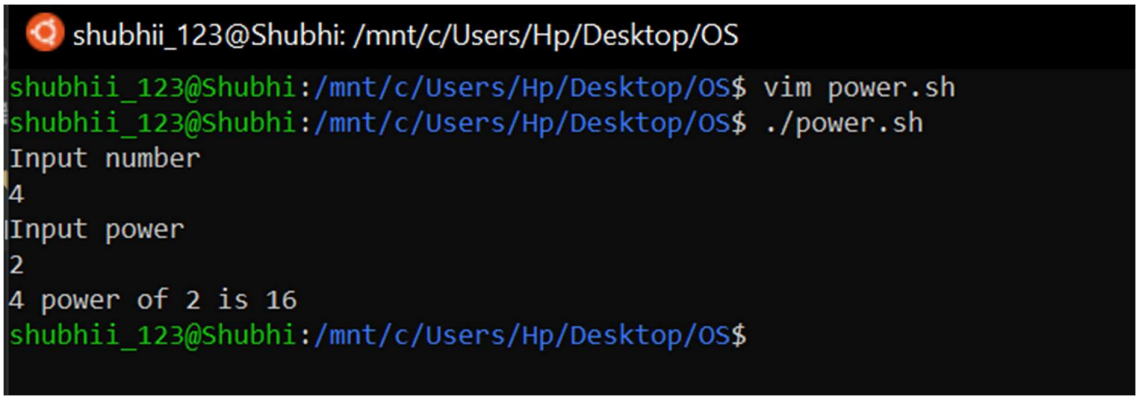
To calculate power of a number raise to the another number

PROGRAM :

shubhii_123@Shubhi: /mnt/c/Users/Hp/Desktop/OS

```
echo "Input number"
read no
echo "Input power"
read power

counter=0
ans=1
while [ $power -ne $counter ]
do
                ans=`expr $ans \* $no`
                        counter=`expr $counter + 1`
                done

                echo "$no power of $power is $ans"

~
~
~
~
~
~
```

OUTPUT:



PROGRAM-6

To count the number of vowels and displaying all of them

PROGRAM:

```
echo
read str
len=$(expr length $str)
count=0
while [ $len -gt 0 ]
do
        ch=$(echo $str | cut -c $len)
        case $ch in
                [aeiouAEIOU] )
                        count=$(($count + 1))
                        echo $ch
                        ;;
        esac
        len=$(( $len - 1 ))
done
echo $count
~
~
~
```

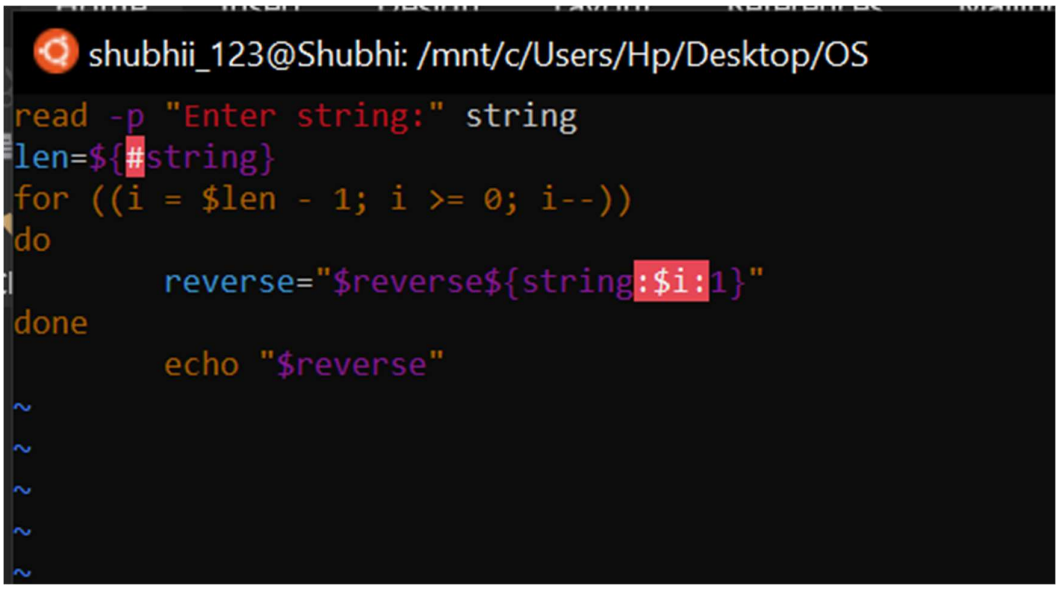
OUTPUT:

```
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/OS$ vim vowels.sh
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/OS$ ./vowels.sh

apple
e
a
2
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/OS$
```

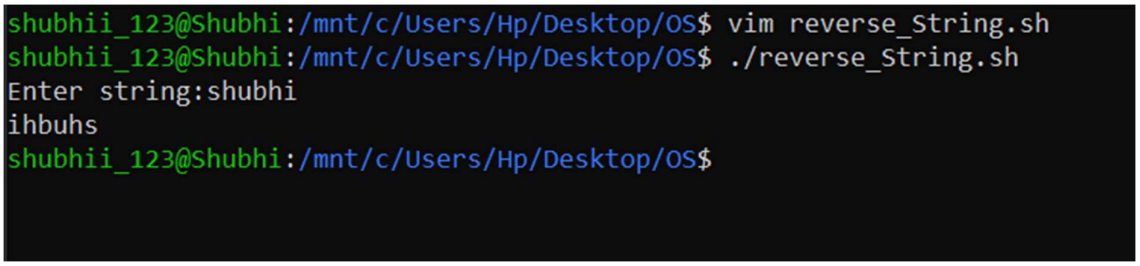
PROGRAM-7

Program to reverse a string

PROGRAM:



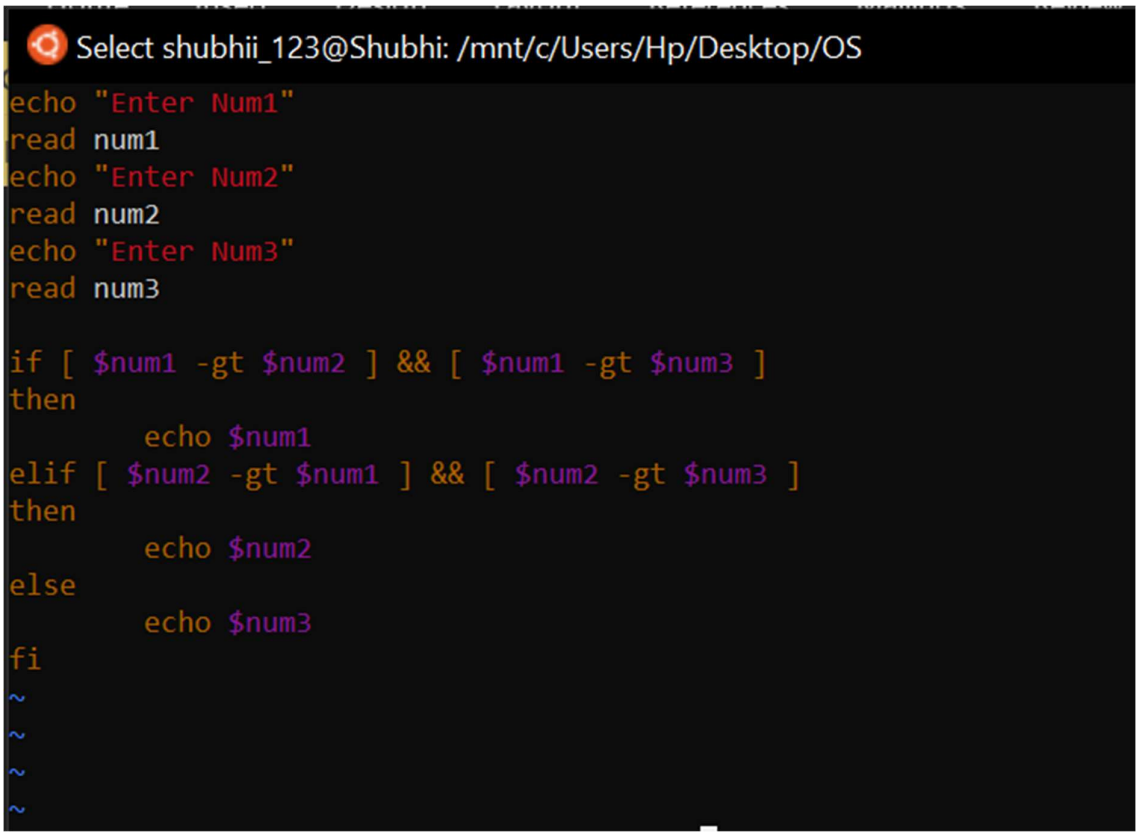
OUTPUT:



PROGRAM-8

To find out greatest of three numbers

PROGRAM:



OUTPUT:

```
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/OS$ vim greatesOf3.sh
shubhii_123@Shubhi:/mnt/c/Users/Hp/Desktop/OS$ ./greatesOf3.sh
Enter Num1
8
Enter Num2
6
Enter Num3
32
32
```

THANK YOU!