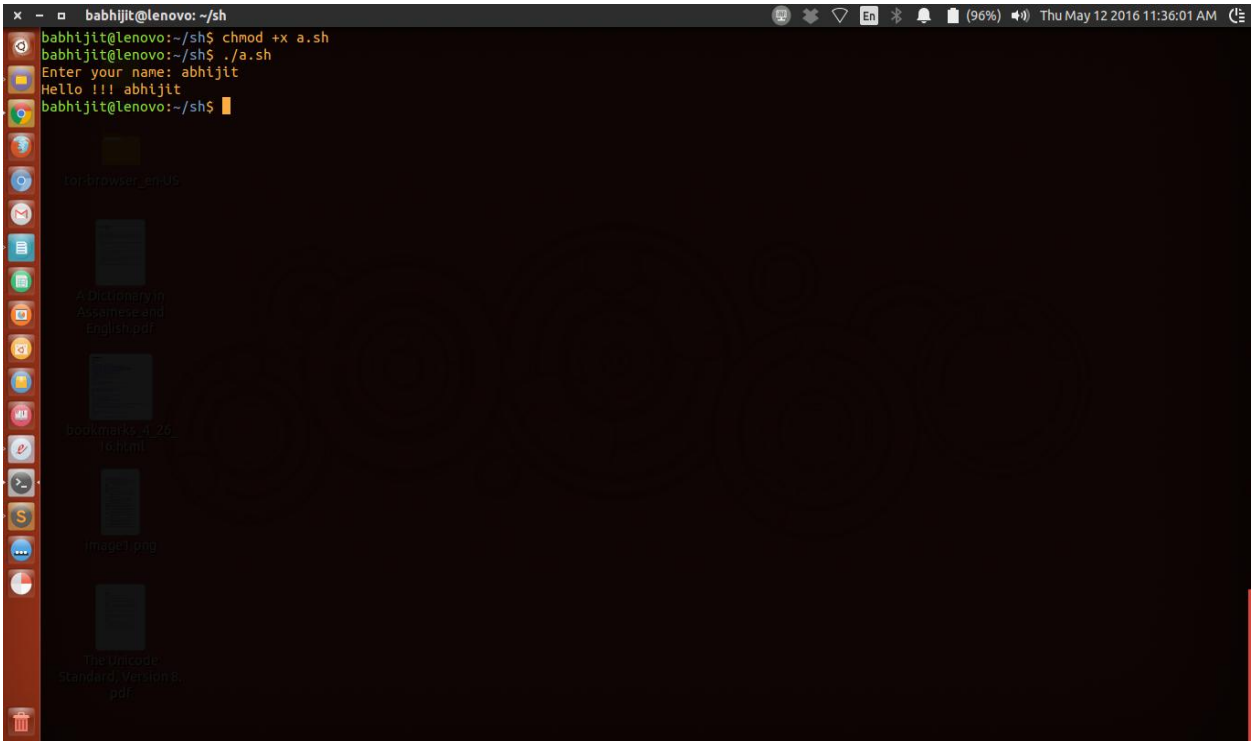
Shell Programming Lab

P1:- Write a shell program to read a string and display it.

Program:-

```
#!/bin/bash
echo -n "Enter your name: ";
read var;
echo "Hello !!! $var";
```

Output:-

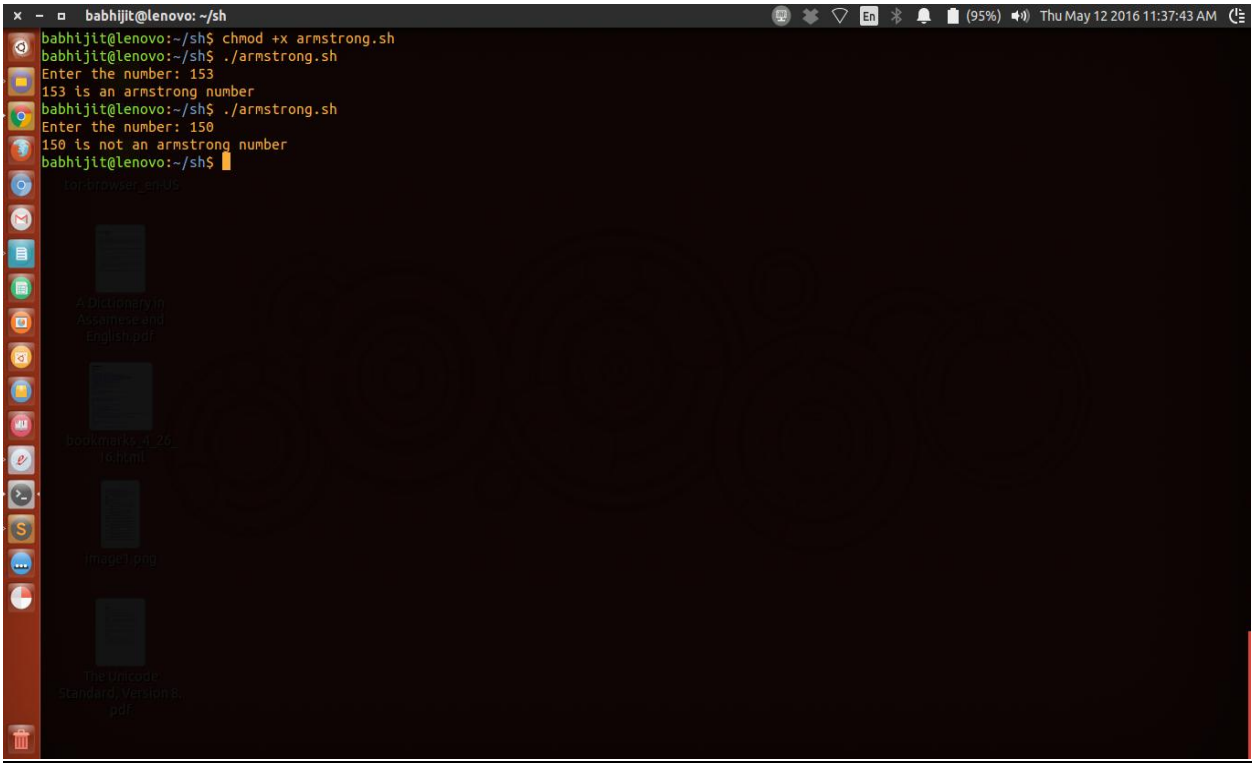


P2:- Write a shell program to check whether a number is armstrong or not.

Program:-

```
#!/bin/sh
echo -n "Enter the number: ";
read num;
q=$num;
res=0;
while [ "$q" -ne 0  ]
do
        re=$(($q%10));
        cube=$(($re \* $re \* $re));
        res=$(($cube+$res));
        q=$(($q/10));
done
if [ "$res" -eq "$num" ]
then
        echo "$num is an armstrong number";
else
        echo "$num is not an armstrong number";
fi
```

Output:-


babhijit@lenovo: ~/sh
babhijit@lenovo:~/sh$ chmod +x armstrong.sh
babhijit@lenovo:~/sh$ ./armstrong.sh
Enter the number: 153
153 is an armstrong number
babhijit@lenovo:~/sh$ ./armstrong.sh
Enter the number: 150
150 is not an armstrong number
babhijit@lenovo:~/sh$
Thu May 12 2016 11:37:43 AM

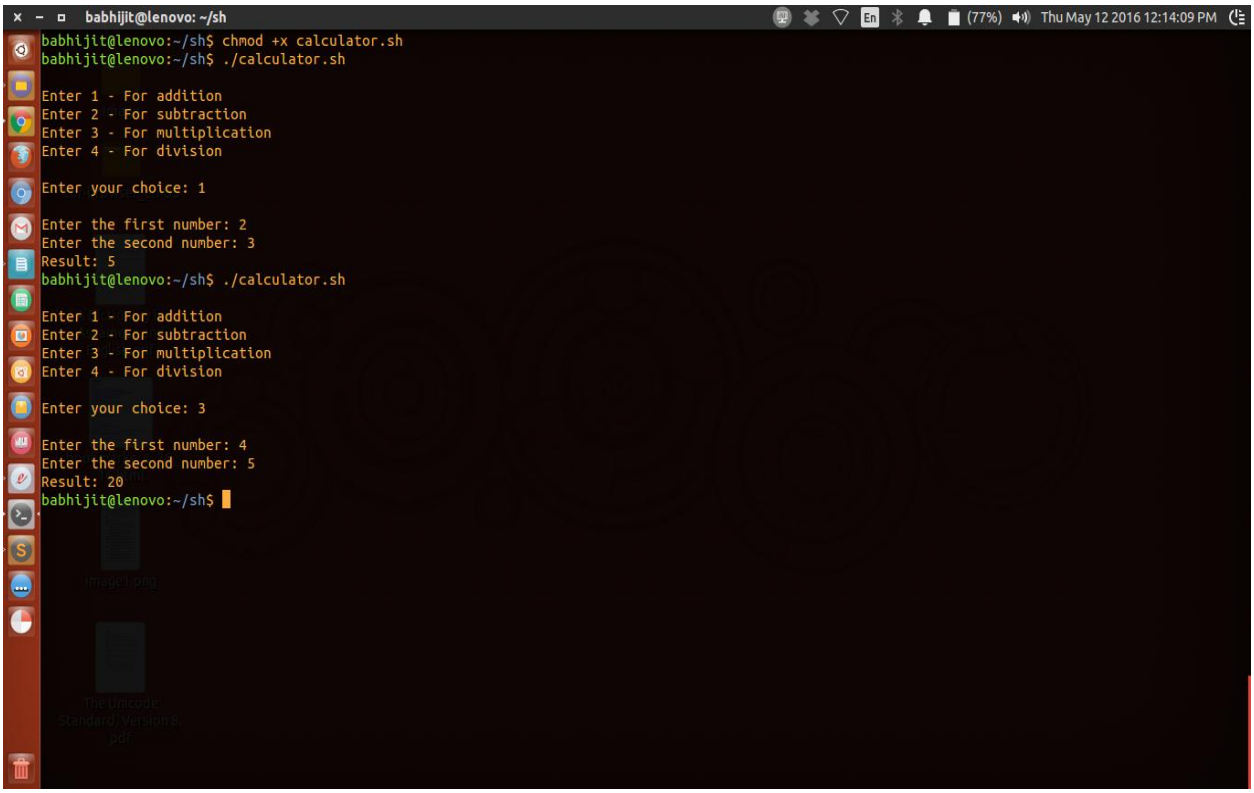
P3:- Write a shell program for a simple calculator.

Program:-

```
#!/bin/sh
echo "\nEnter 1 - For addition";
echo "Enter 2 - For subtraction";
echo "Enter 3 - For multiplication";
echo "Enter 4 - For division";
echo -n "\nEnter your choice: ";
read n;
echo -n "\nEnter the first number: ";
read num1;
echo -n "Enter the second number: ";
read num2;
case "$n" in
        "1")    #echo "Result: $(($num1+$num2)) ";
                echo -n "Result: "; awk "BEGIN {print $num1+$num2;exit}";
        ;;
        "2")    #echo "Result: $(($num1-$num2))";
                echo -n "Result: "; awk "BEGIN {print $num1-$num2;exit}";
        ;;
        "3")    #echo "Result: $(($num1*$num2))";
                echo -n "Result: "; awk "BEGIN {print $num1*$num2;exit}";
        ;;
```

```
        "4")    #echo "Result: $(($num1/$num2))";
                echo -n "Result: "; awk "BEGIN {print $num1/$num2;exit}";
        ;;
esac
```

Output:-

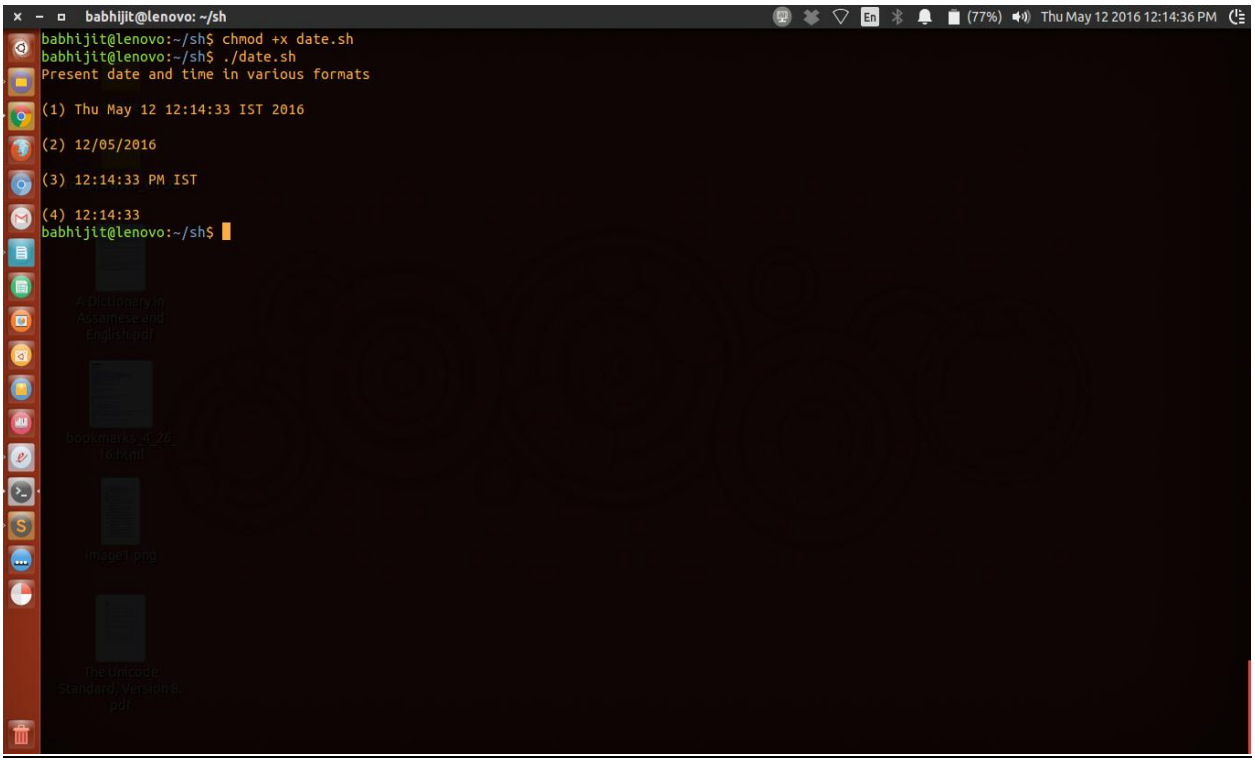


P4:- Write a shell program to display various date and time formats.

Program:-

```
#!/bin/sh
echo "Present date and time in various formats";
echo -n "\n(1) ";
date
echo -n "\n(2) ";
date +'%d/%m/%Y'
echo -n "\n(3) ";
date +'%r'
echo -n "\n(4) ";
date +'%I:%M:%S'
```

Output:-

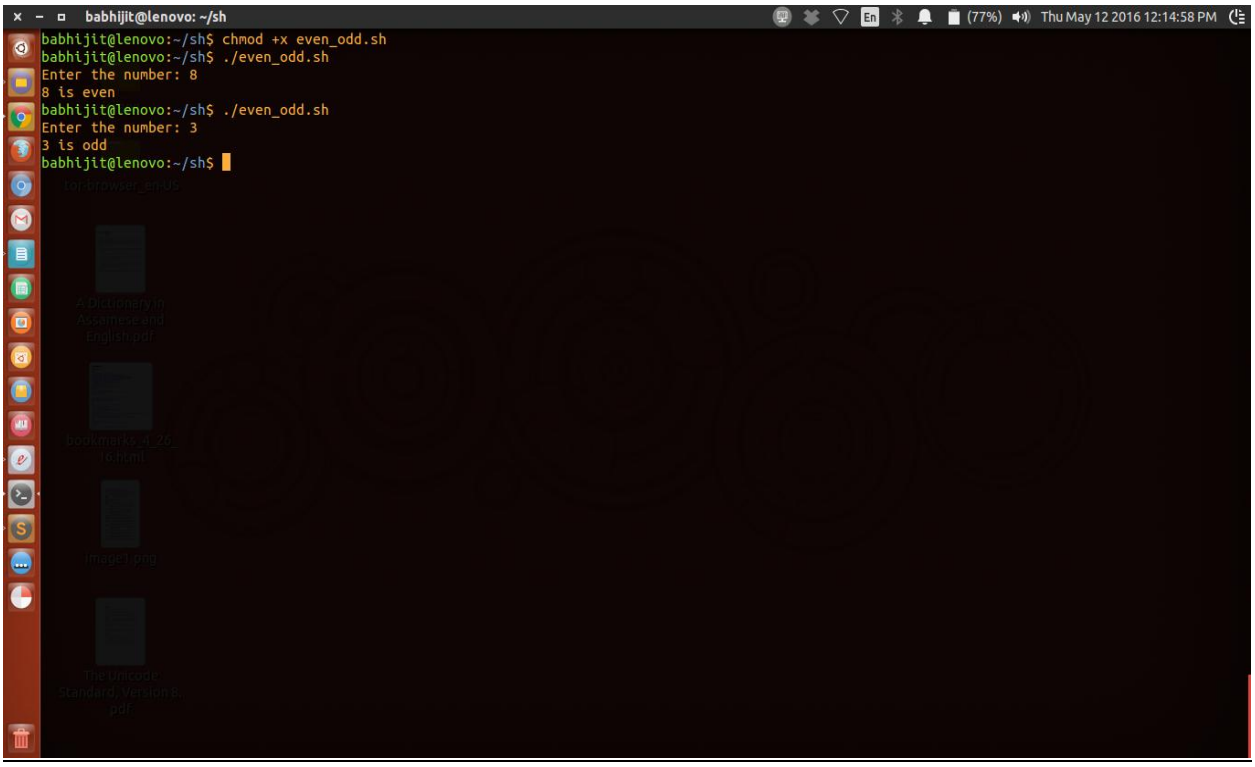


P5:- Write a shell program to check whether a number is even or not.

Program:-

```
#!/bin/sh
echo -n "Enter the number: "
read var
#r=$(($var%2))
if [ "$(($var%2))" -eq 0 ]
then
        echo "$var is even";
else
        echo "$var is odd";
fi
```

Output:-

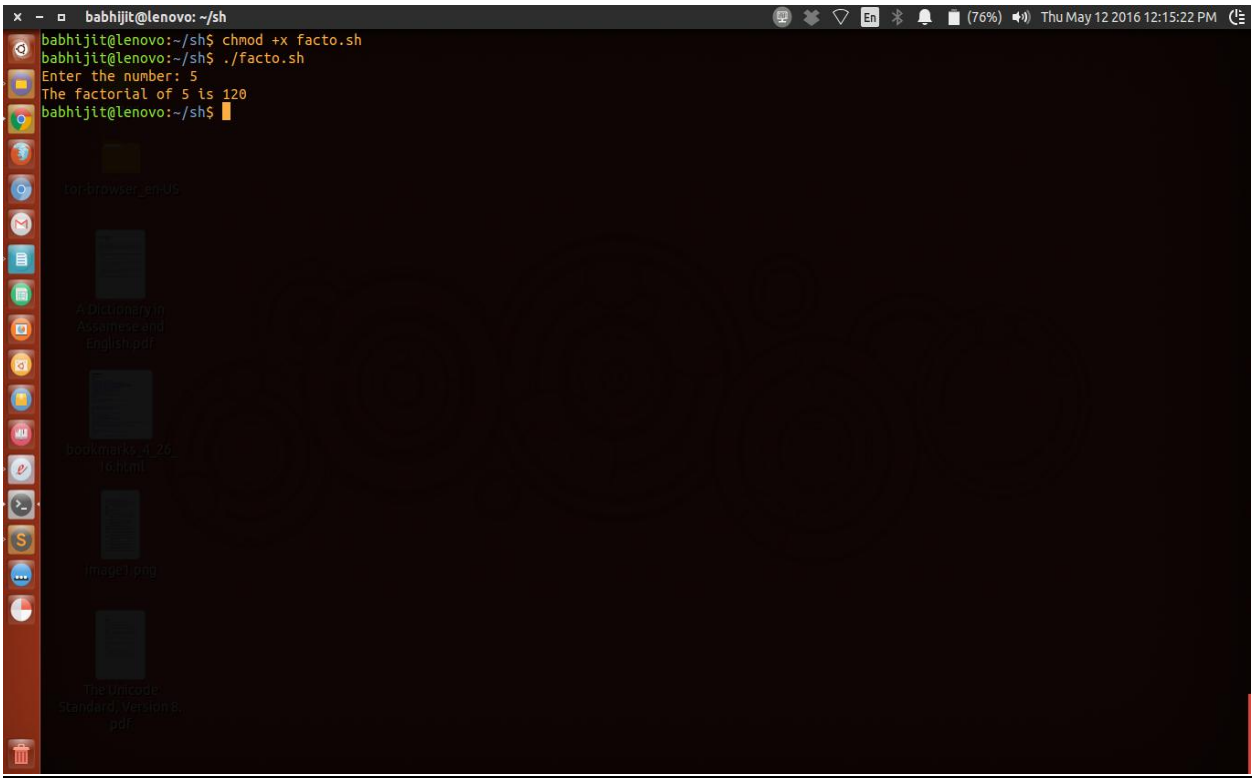


P6:- Write a shell program to find the factorial of a number.

Program:-

```
#!/bin/sh
echo -n "Enter the number: "
read n
fact=1;
for i in $(bash -c "echo {2..${n}}")
do
        fact=$(($fact \* $i));
done
echo "The factorial of $n is $fact";
```

Output:-

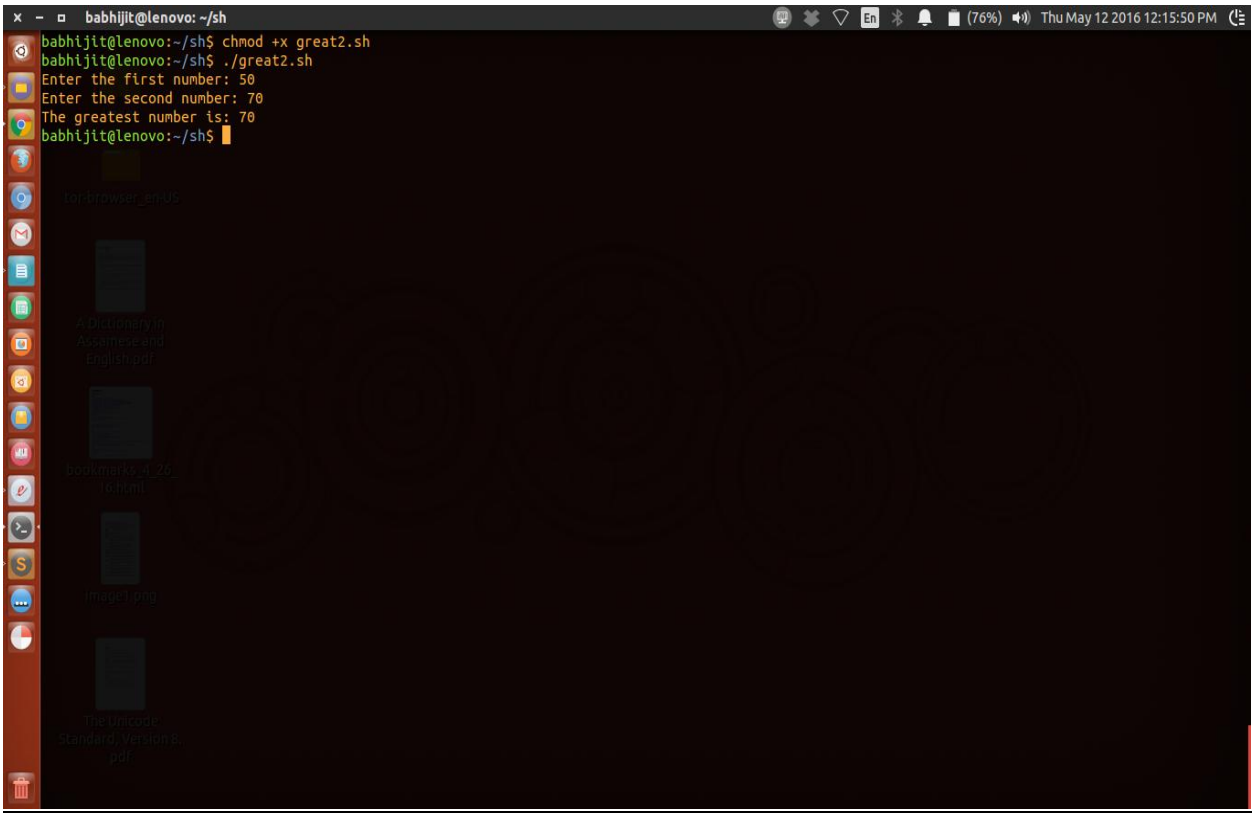


P7:- Write a shell program to find the greatest of 2 numbers.

Program:-

```
#!/bin/sh
echo -n "Enter the first number: ";
read a;
echo -n "Enter the second number: ";
read b;
grt=$([ "$a" -gt "$b" ] && echo "$a" || echo "$b");
echo "The greatest number is: $grt";
```

Output:-

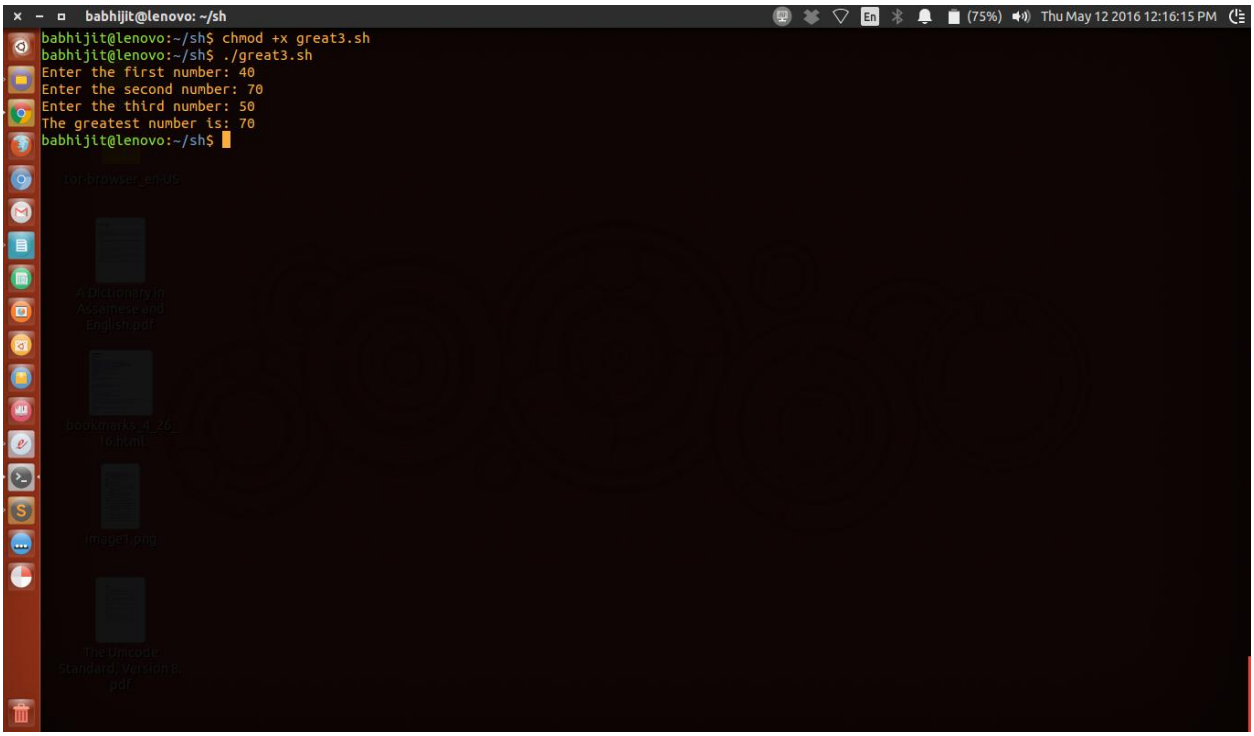


P8:- Write a shell program to find the greatest of 3 numbers.

Program:-

```
#!/bin/sh
echo -n "Enter the first number: ";
read a;
echo -n "Enter the second number: ";
read b;
echo -n "Enter the third number: ";
read c;
#grt=$([ "$a" -gt "$b" ] && echo "$a" || echo "$b");
grt=$([ "$a" -gt "$b" -a "$a" -gt "$c" ] && echo "$a" || ([ "$b" -gt "$c" ] && echo "$b" || echo "$c" ));
echo "The greatest number is: $grt";
```

Output:-



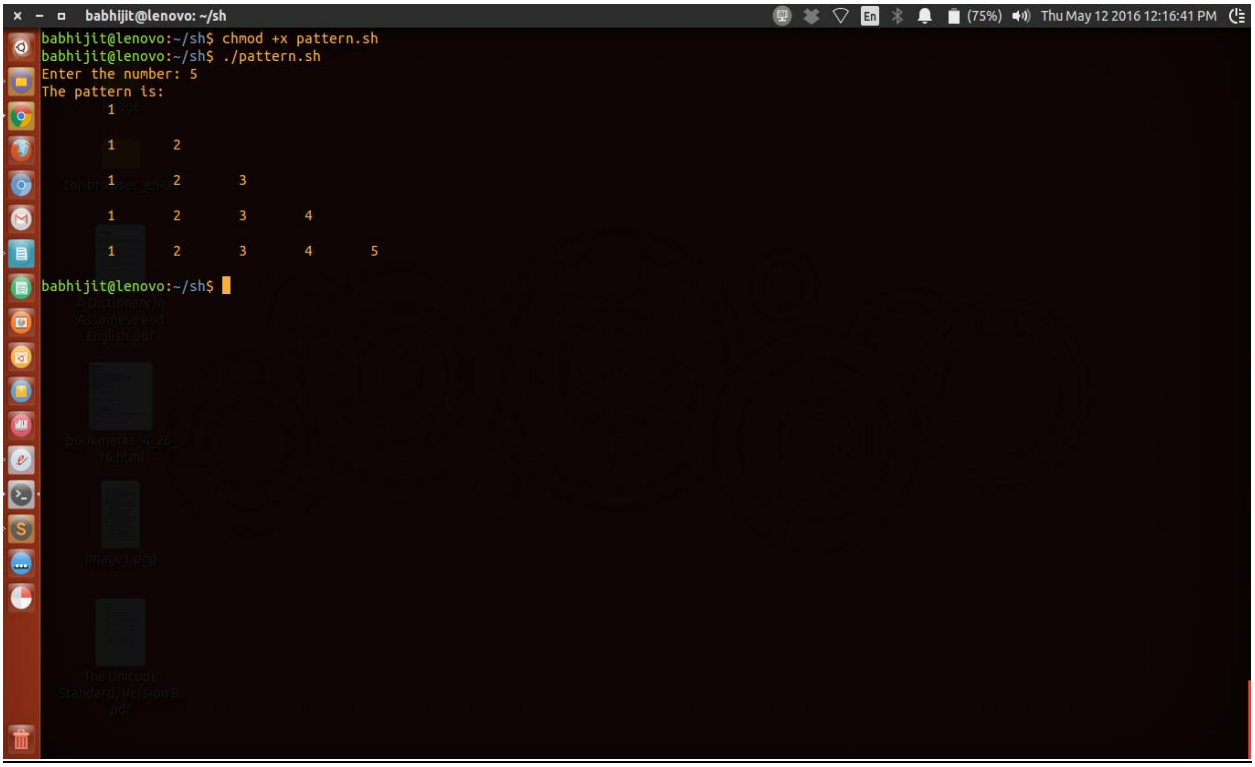
P9:- Write a shell program to generate the following pattern –

```
1
1    2
1    2    3
1    2    3    4
1    2    3    4    5
```

Program:-

```
#!/bin/sh
echo -n "Enter the number: ";
read n;
echo "The pattern is: ";
for i in $(bash -c "echo {1..${n}}")
do
        for j in $(bash -c "echo {1..${i}}")
        do
                echo -n "\t$j";
        done
        echo "\n";
done
```

Output:-


babhijit@lenovo: ~/sh
babhijit@lenovo:~/sh$ chmod +x pattern.sh
babhijit@lenovo:~/sh$ ./pattern.sh
Enter the number: 5
The pattern is:
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
babhijit@lenovo:~/sh$

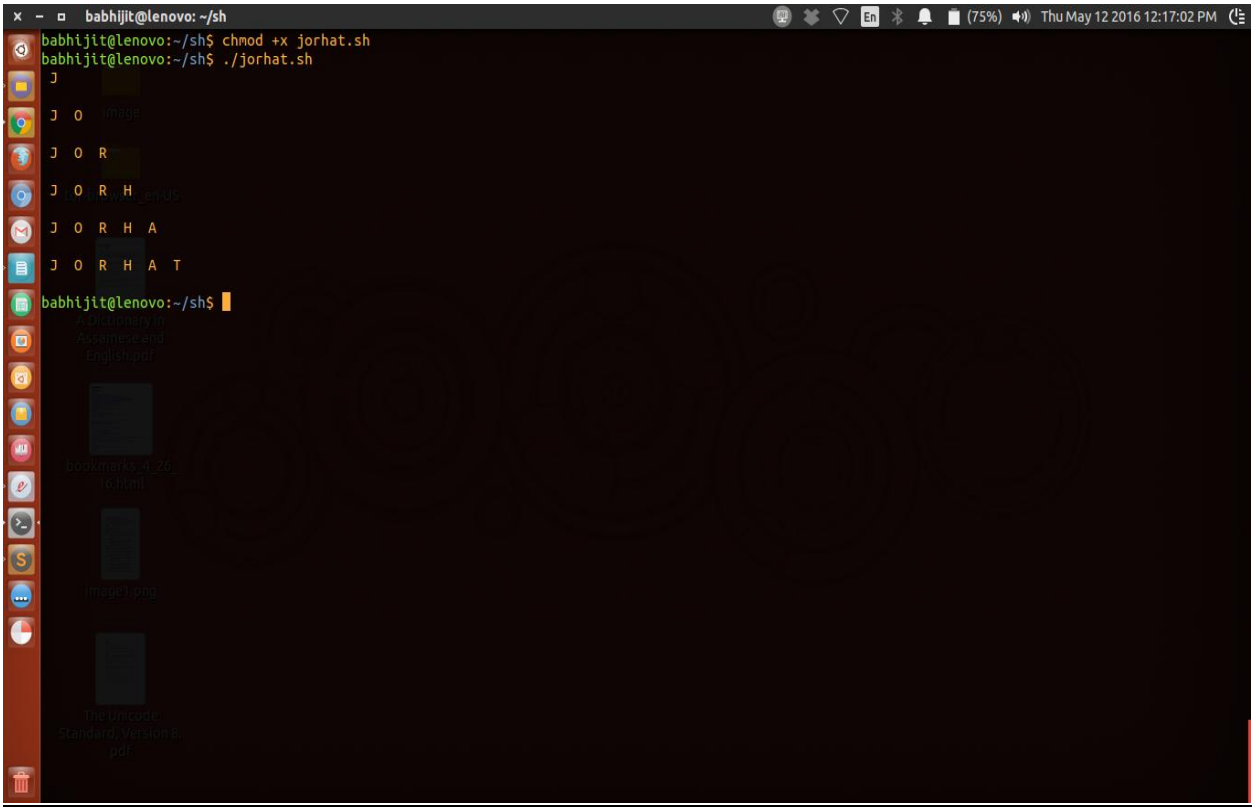
P10:- Write a shell program to generate the following pattern.

```
J
J       O
J       O       R
J       O       R       H
J       O       R       H       A
J       O       R       H       A       T
```

Program:-

```
#!/bin/bash
#ar[0]="J";
#ar[1]="O";
#ar[2]="R";
#ar[3]="H";
#ar[4]="A";
#ar[5]="T";
ar=("J" "O" "R" "H" "A" "T");
for i in $(bash -c "echo {0..5}")
do
        for j in $(bash -c "echo {0..${i}}")
        do
                echo -n " ${ar[$j]} ";
                #awk "BEGIN {print $j;exit}";
        done
        echo -e "\n";
done
```

Output:-

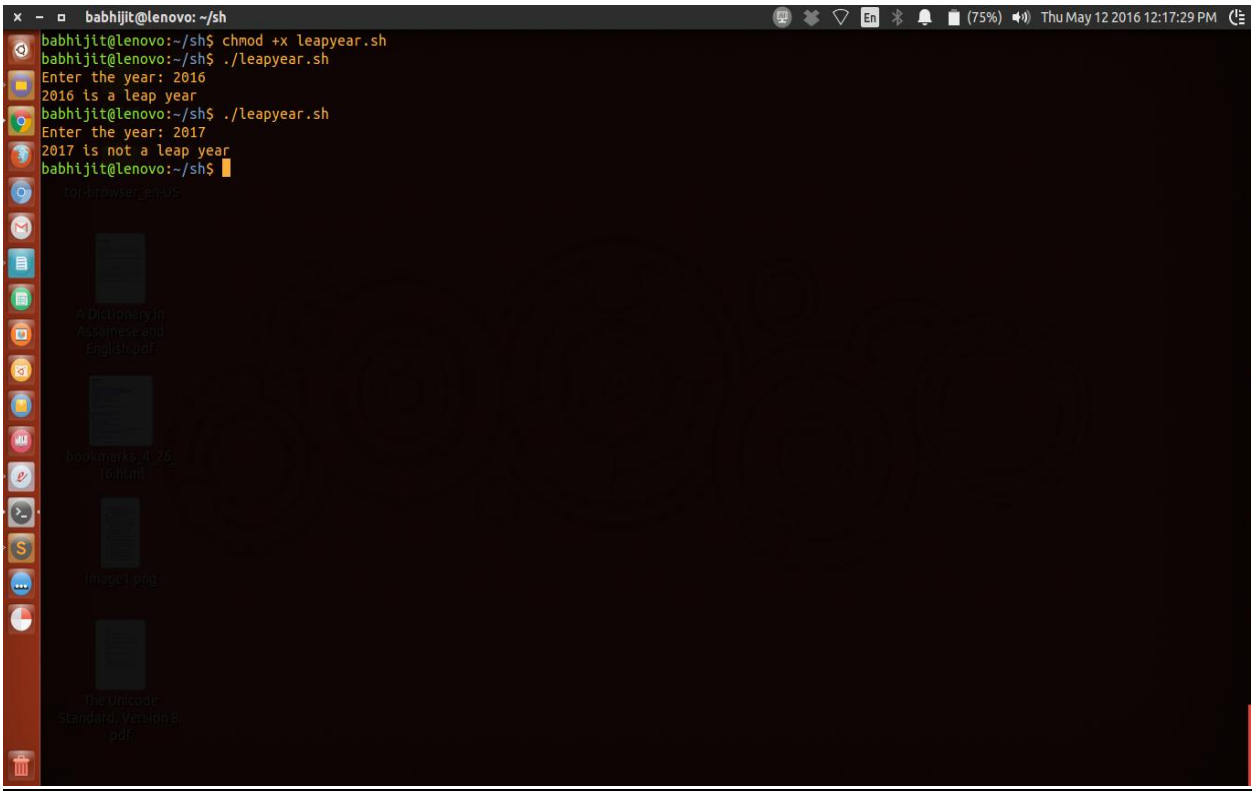

babhijit@lenovo: ~/sh
babhijit@lenovo:~/sh$ chmod +x jorhat.sh
babhijit@lenovo:~/sh$ ./jorhat.sh
J
J O
J O R
J O R H
J O R H A
J O R H A T
babhijit@lenovo:~/sh$
(75%)
Thu May 12 2016 12:17:02 PM

P11:- Write a shell program to check whether a year is leap year or not.

Program:-

```
#!/bin/sh
echo -n "Enter the year: ";
read num;
if [ $(($num%4)) -eq 0 ]
then
        echo "$num is a leap year";
else
        echo "$num is not a leap year";
fi
```

Output:-

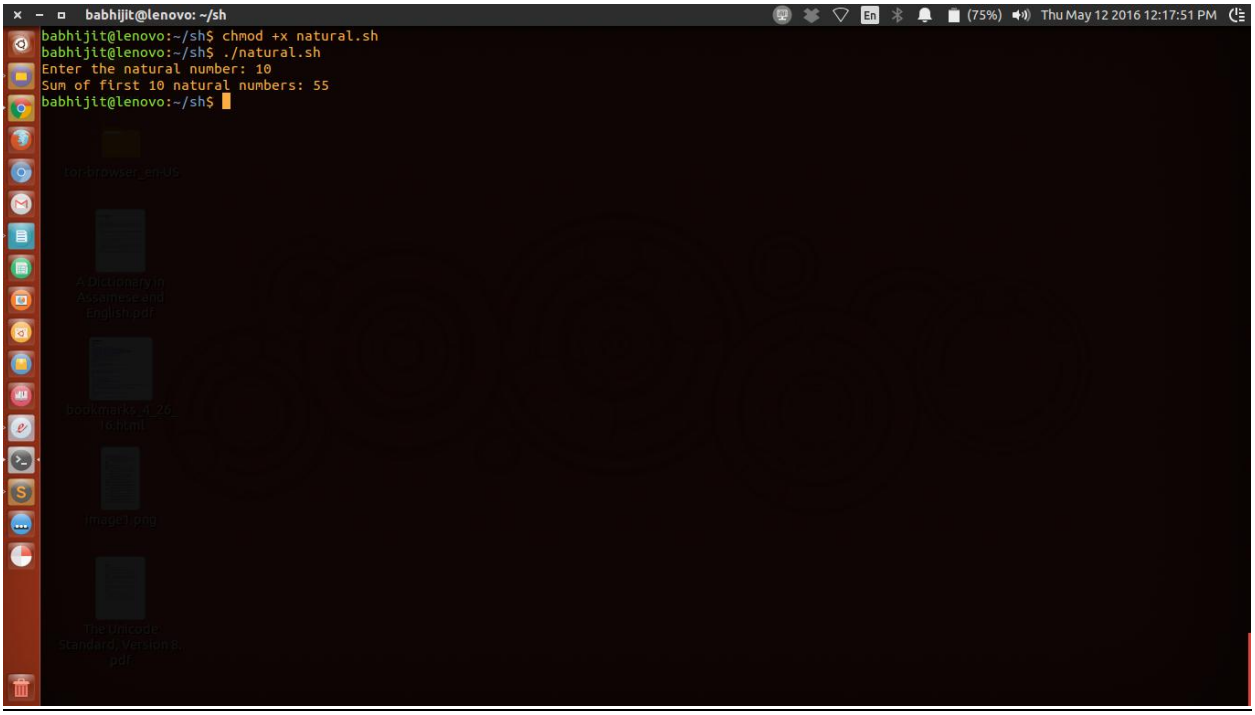


P12:- Write a shell program to find the sum of first ‘n’ natural numbers.

Program:-

```
#/bin/sh
echo -n "Enter the natural number: ";
read var;
count=0;
sumn=0;
while [ "$var" -ge "$count" ]
do
        sumn=$(($sumn+$count));
        count=$(($count+1));
done
echo "Sum of first $var natural numbers: $sumn";
```

Output:-

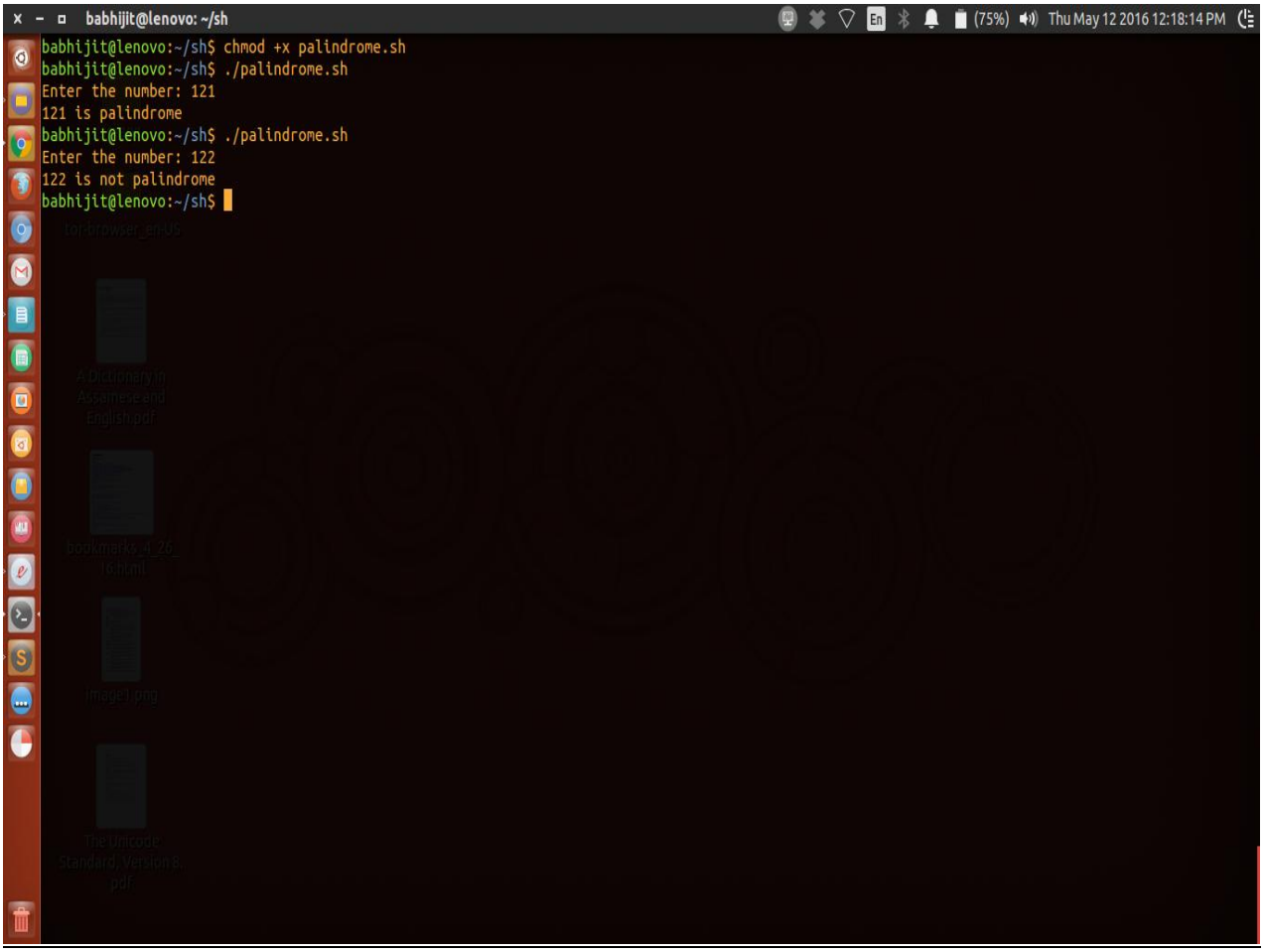


P13:- Write a shell program to check whether a number is palindrome or not.

Program:-

```
#!/bin/sh
echo -n "Enter the number: ";
read n;
m=$n;
res=0;
while [ "$n" -ne 0  ]
do
        re=$(($n%10));
        res=$((10 \* $res + re));
        n=$(($n/10));
done
if [ "$m" -eq "$res" ]
then
        echo "$m is palindrome";
else
        echo "$m is not palindrome";
fi
```

Output:-

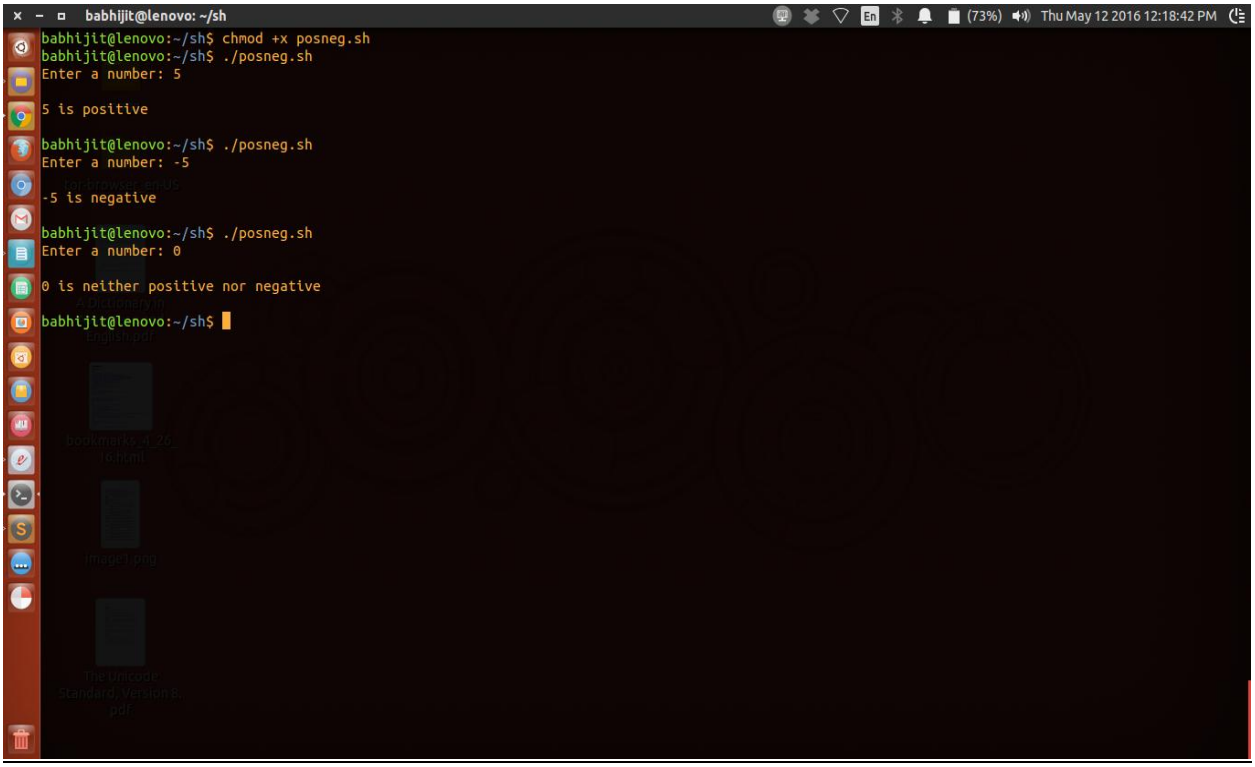

babhijit@lenovo: ~/sh
Thu May 12 2016 12:18:14 PM
babhijit@lenovo:~/sh$ chmod +x palindrome.sh
babhijit@lenovo:~/sh$ ./palindrome.sh
Enter the number: 121
121 is palindrome
babhijit@lenovo:~/sh$ ./palindrome.sh
Enter the number: 122
122 is not palindrome
babhijit@lenovo:~/sh$

P14:- Write a shell program to check whether a number is positive or negative.

Program:-

```
#!/bin/sh
echo -n "Enter a number: ";
read num;
if [ "$num" -gt 0 ]
then
        echo "\n$num is positive\n";
fi
if [ "$num" -lt 0 ]
then
        echo "\n$num is negative\n";
fi
if [ "$num" -eq 0 ]
then
        echo "\n$num is neither positive nor negative\n";
fi
```

Output:-

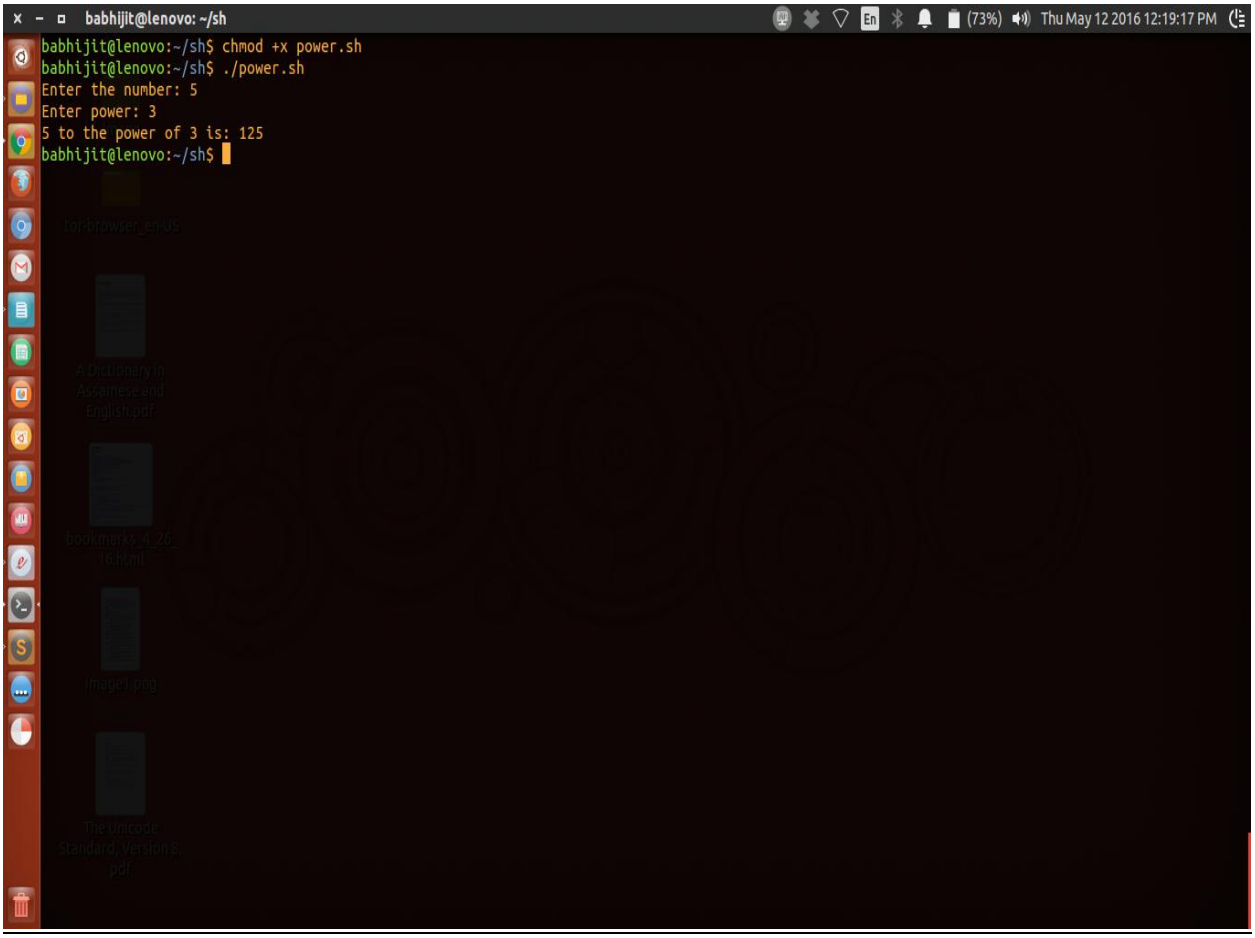

babhijit@lenovo: ~/sh
babhijit@lenovo:~/sh$ chmod +x posneg.sh
babhijit@lenovo:~/sh$ ./posneg.sh
Enter a number: 5
5 is positive
babhijit@lenovo:~/sh$ ./posneg.sh
Enter a number: -5
-5 is negative
babhijit@lenovo:~/sh$ ./posneg.sh
Enter a number: 0
0 is neither positive nor negative
babhijit@lenovo:~/sh$
Thu May 12 2016 12:18:42 PM

P15:- Write a shell program to find the power of a number (x^y).

Program:-

```
#!/bin/sh
echo -n "Enter the number: ";
read x;
echo -n "Enter power: ";
read y;
res=1;
if [ "$y" -eq 0 ]
then
        res=1;
else
for i in $(bash -c "echo {1..${y}}")
do
        res=$(($res \* $x));
done
fi
#res=$(($a \* $a \* $a));
echo "$x to the power of $y is: $res";
```

Output:-

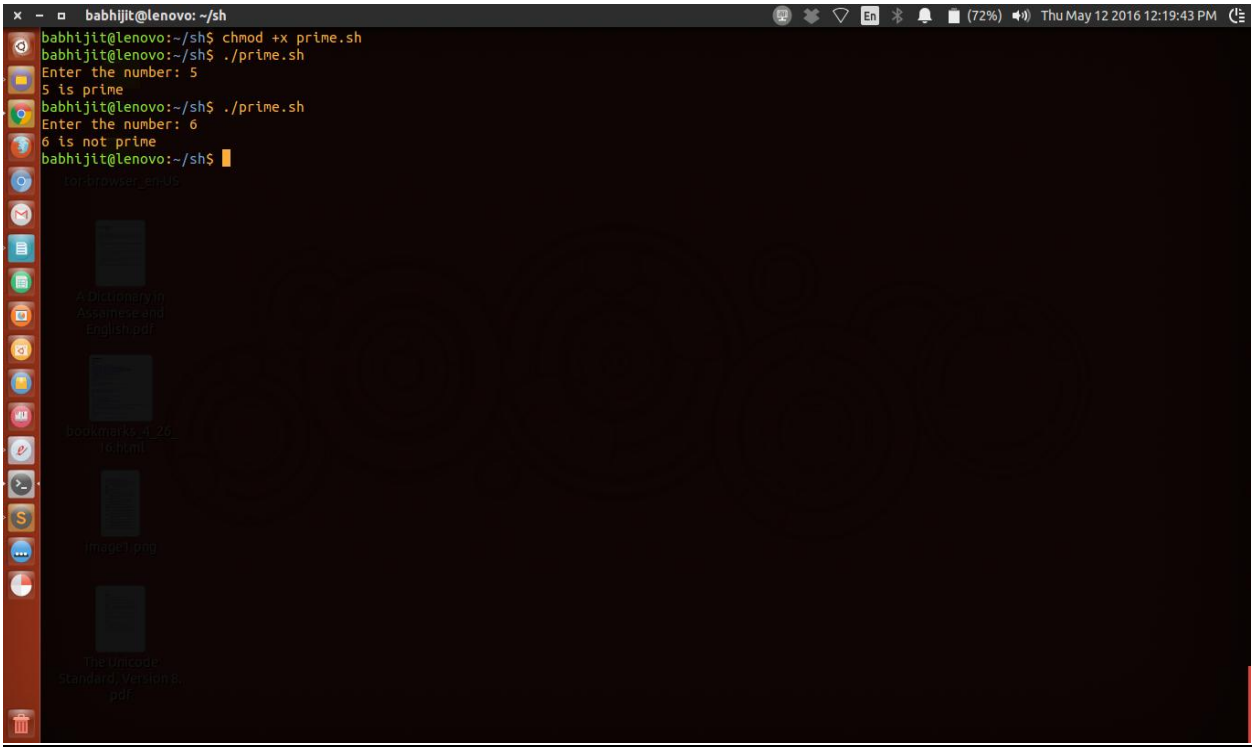

babhijit@lenovo: ~/sh
(73%) Thu May 12 2016 12:19:17 PM
babhijit@lenovo:~/sh$ chmod +x power.sh
babhijit@lenovo:~/sh$ ./power.sh
Enter the number: 5
Enter power: 3
5 to the power of 3 is: 125
babhijit@lenovo:~/sh$

P16:- Write a shell program to check whether a number is prime or not.

Program:-

```
#!/bin/sh
echo -n "Enter the number: ";
read n
for i in $(bash -c "echo {2..${n}}")
do
        #echo "$i"
        if [ "$(($n%$i))" -eq 0 ]
        then
                break;
        fi
done
if [ "$n" -eq "$i" ]
then
        echo "$n is prime";
else
        echo "$n is not prime";
fi
```

Output:-

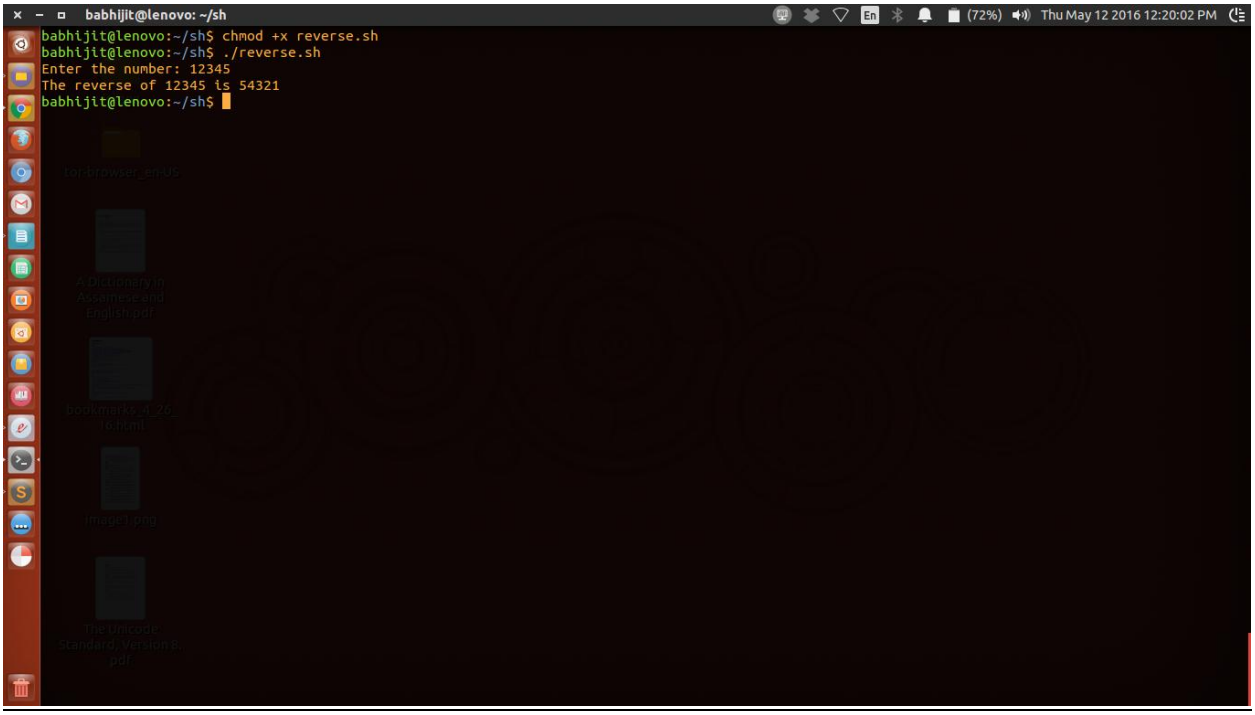

babhijit@lenovo: ~/sh
babhijit@lenovo:~/sh$ chmod +x prime.sh
babhijit@lenovo:~/sh$ ./prime.sh
Enter the number: 5
5 is prime
babhijit@lenovo:~/sh$ ./prime.sh
Enter the number: 6
6 is not prime
babhijit@lenovo:~/sh$
Thu May 12 2016 12:19:43 PM

P17:- Write a shell program to reverse a number.

Program:-

```
#!/bin/sh
echo -n "Enter the number: ";
read n;
m=$n;
res=0;
while [ "$n" -ne 0  ]
do
        re=$(($n%10));
        res=$((10 \* $res + re));
        n=$(($n/10));
done
echo "The reverse of $m is $res";
```

Output:-

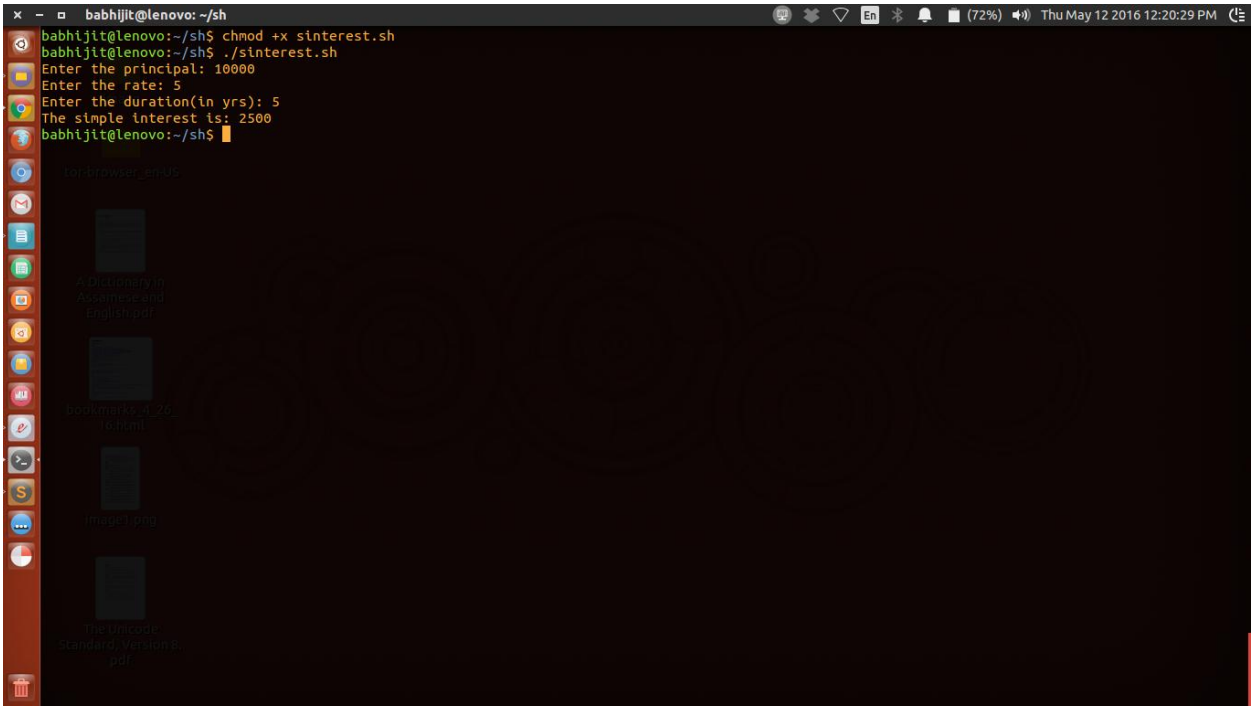


P18:- Write a shell program to calculate simple interest.

Program:-

```
#!/bin/sh
echo -n "Enter the principal: ";
read p;
echo -n "Enter the rate: ";
read r;
echo -n "Enter the duration(in yrs): ";
read n;
si=0;
x=$((p \* r \* n));
si=$(($x/100));
echo "The simple interest is: $si";
```

Output:-

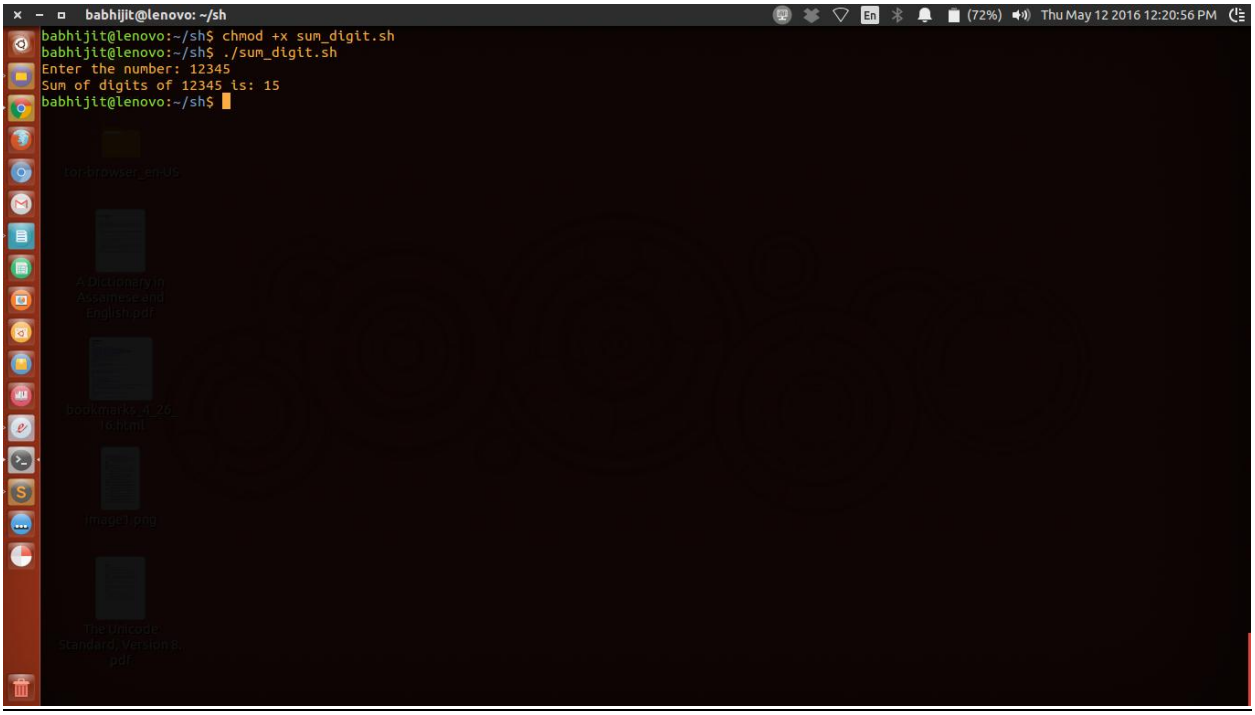


P19:- Write a shell program to find the sum of the digits of a number.

Program:-

```
#!/bin/sh
echo -n "Enter the number: ";
read n;
m=$n;
while [ "$n" -ne 0 ]
do
        re=$(($n%10));
        res=$(($res+$re));
        n=$(($n/10));
done
echo "Sum of digits of $m is: $res";
```

Output:-

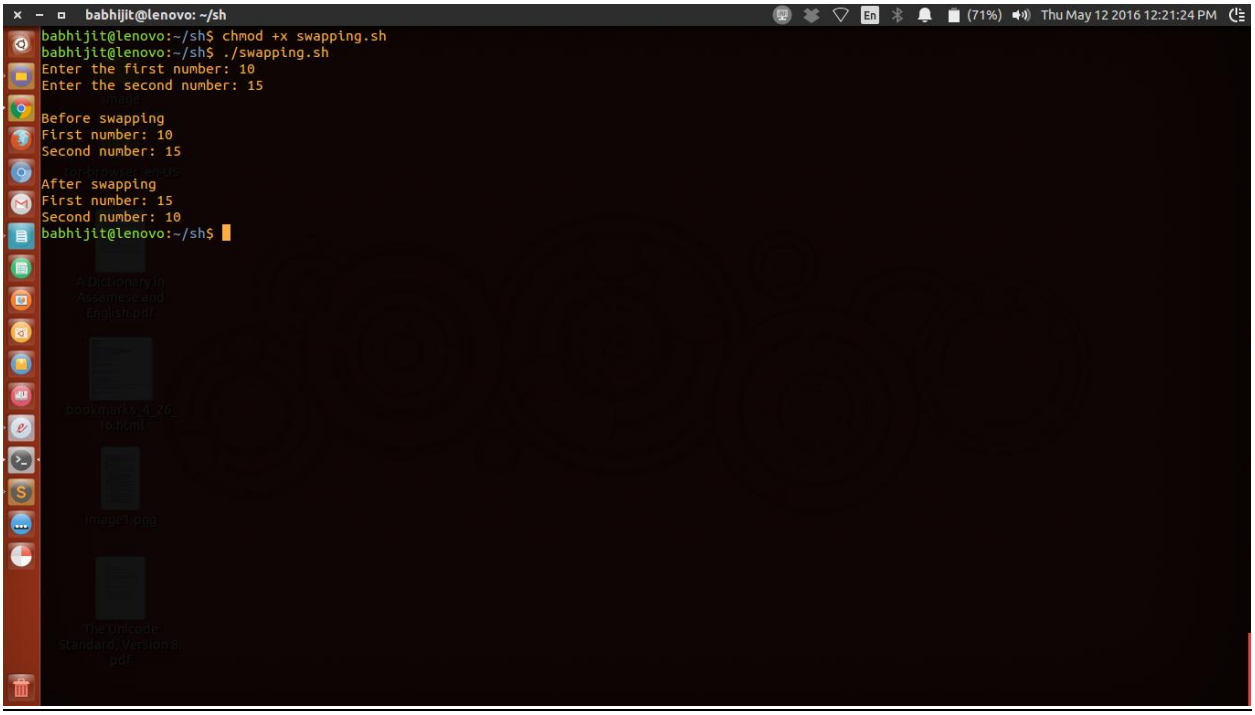


P20:- Write a shell program to swap 2 numbers.

Program:-

```
#!/bin/sh
echo -n "Enter the number: ";
read n;
m=$n;
while [ "$n" -ne 0 ]
do
        re=$(($n%10));
        res=$(($res+$re));
        n=$(($n/10));
done
echo "Sum of digits of $m is: $res";
```

Output:-



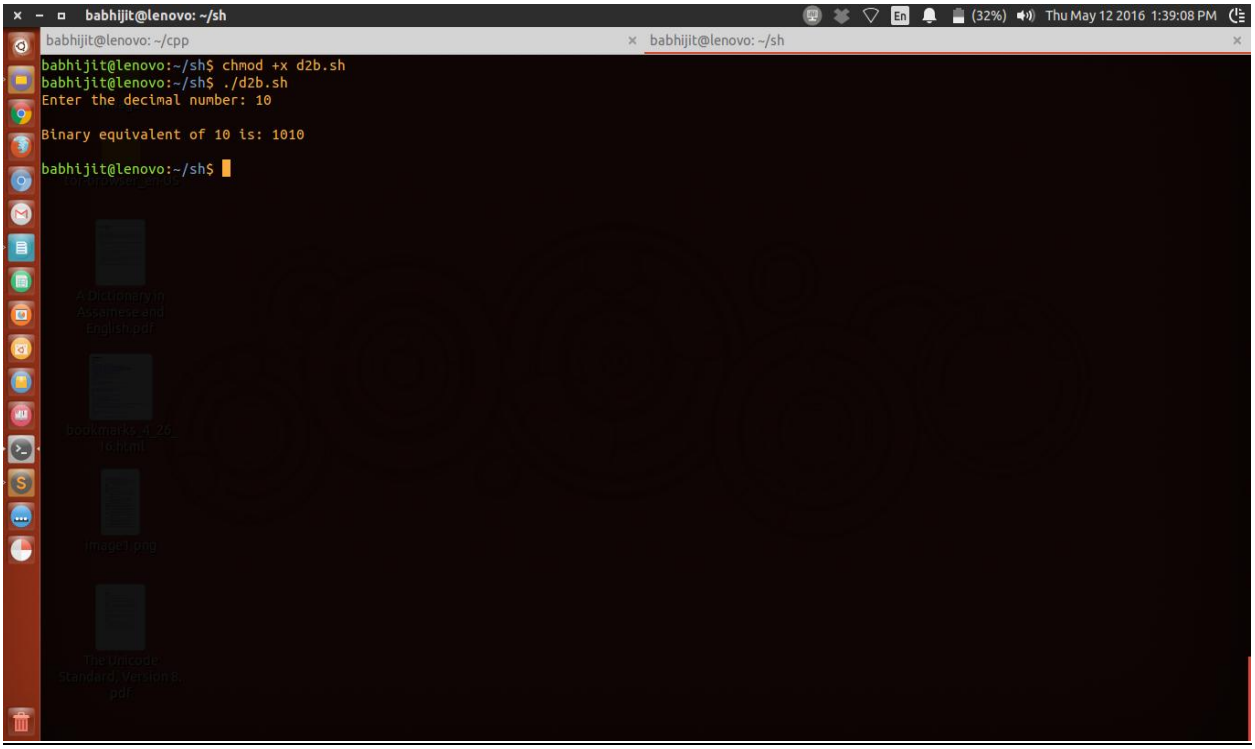
P21:- Write a shell program to convert a decimal number to equivalent binary number.

Program:-

```
#!/bin/sh
echo -n "Enter the decimal number: ";
read num;
n=$num;
p=0;
res=0;
x=0;
k=0;
flag=0;
while [ "$n" -gt 0 ]
do
        r=$(($n%2));
        if [ "$r" -eq 0 -a "$flag" -eq 0 ]
        then
                k=$(($k+1));
        else
                flag=1;
        fi
        x=$((res \* 10));
        res=$(($r+$x));
        n=$(($n/2));
done
n=$res;
```

```
res=0;
x=0;
while [ "$n" -gt 0 ]
do
        r=$(($n%10));
        x=$(($res \* 10));
        res=$(($r+$x));
        n=$(($n/10));
done
if [ "$k" -gt 0 ]
then
        for i in $(bash -c "echo {1..${k}}")
        do
                res=$(($res \* 10));
        done
fi
echo "\nBinary equivalent of $num is: $res\n";
```

Output:-

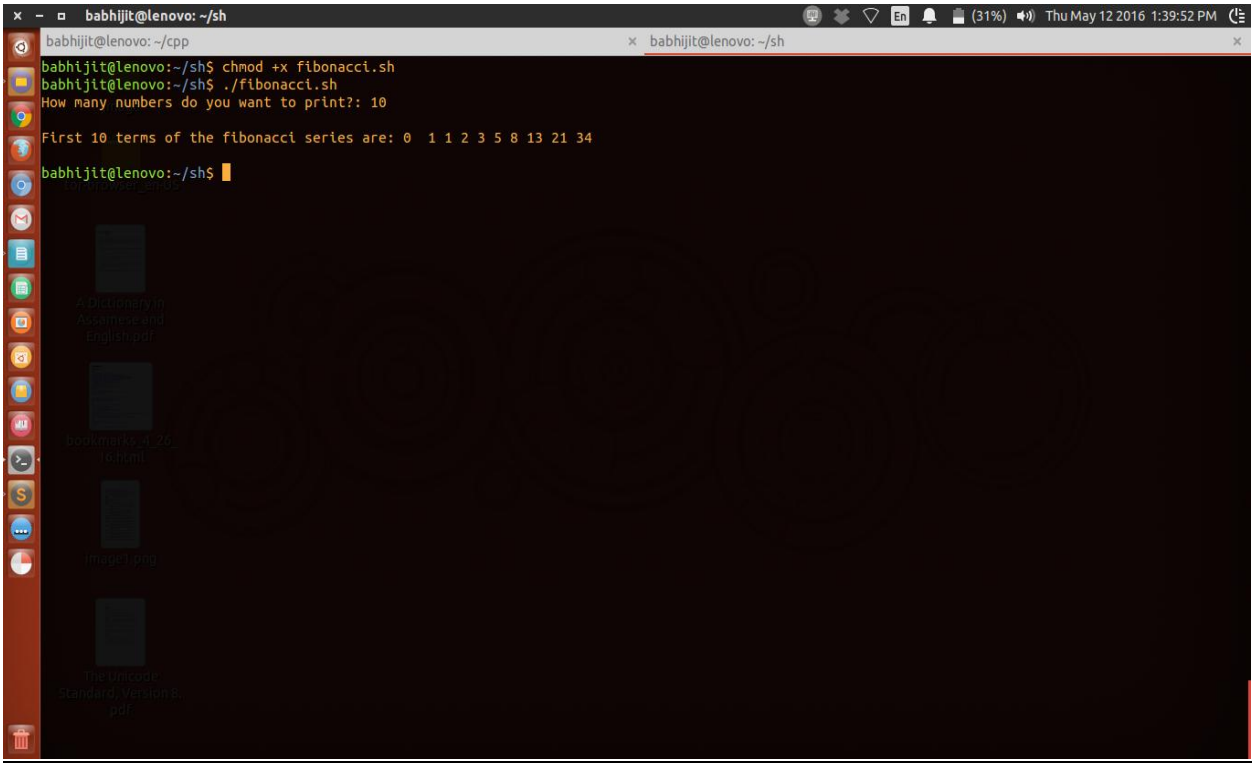

babhijit@lenovo: ~/sh
babhijit@lenovo: ~/cpp
babhijit@lenovo: ~/sh
babhijit@lenovo:~/sh$ chmod +x d2b.sh
babhijit@lenovo:~/sh$ ./d2b.sh
Enter the decimal number: 10
Binary equivalent of 10 is: 1010
babhijit@lenovo:~/sh$

P22:- Write a shell program to generate the Fibonacci series.

Program:-

```
#!/bin/sh
echo -n "How many numbers do you want to print?: ";
read num;
fterm=0;
sterm=1;
nterm=0;
count=2;
echo -n "\nFirst $num terms of the fibonacci series are: ";
echo -n "$fterm  $sterm ";
while [ "$count" -lt "$num" ]
do
        nterm=$(($fterm+$sterm));
        echo -n "$nterm ";
        fterm=$sterm;
        sterm=$nterm;
        count=$(($count+1));
done
echo "\n";
```

Output:-


babhijit@lenovo: ~/sh
babhijit@lenovo: ~/cpp
babhijit@lenovo: ~/sh
babhijit@lenovo:~/sh$ chmod +x fibonacci.sh
babhijit@lenovo:~/sh$ ./fibonacci.sh
How many numbers do you want to print?: 10
First 10 terms of the fibonacci series are: 0 1 1 2 3 5 8 13 21 34
babhijit@lenovo:~/sh$

P23:- Write a shell program to read a number of strings and to display them.

Program:-

```
#!/bin/sh
echo -n "What is your name?: ";
read name
clear
echo "Hello $name";
echo -n "What is your favorite color?: ";
read color
clear
echo "$color is a good color";
echo "Now saving that info";
echo "Favorite color of $name is $color";
echo "Data saved";
echo "Press enter to continue";
read
clear
echo "Have a good day $name";
```

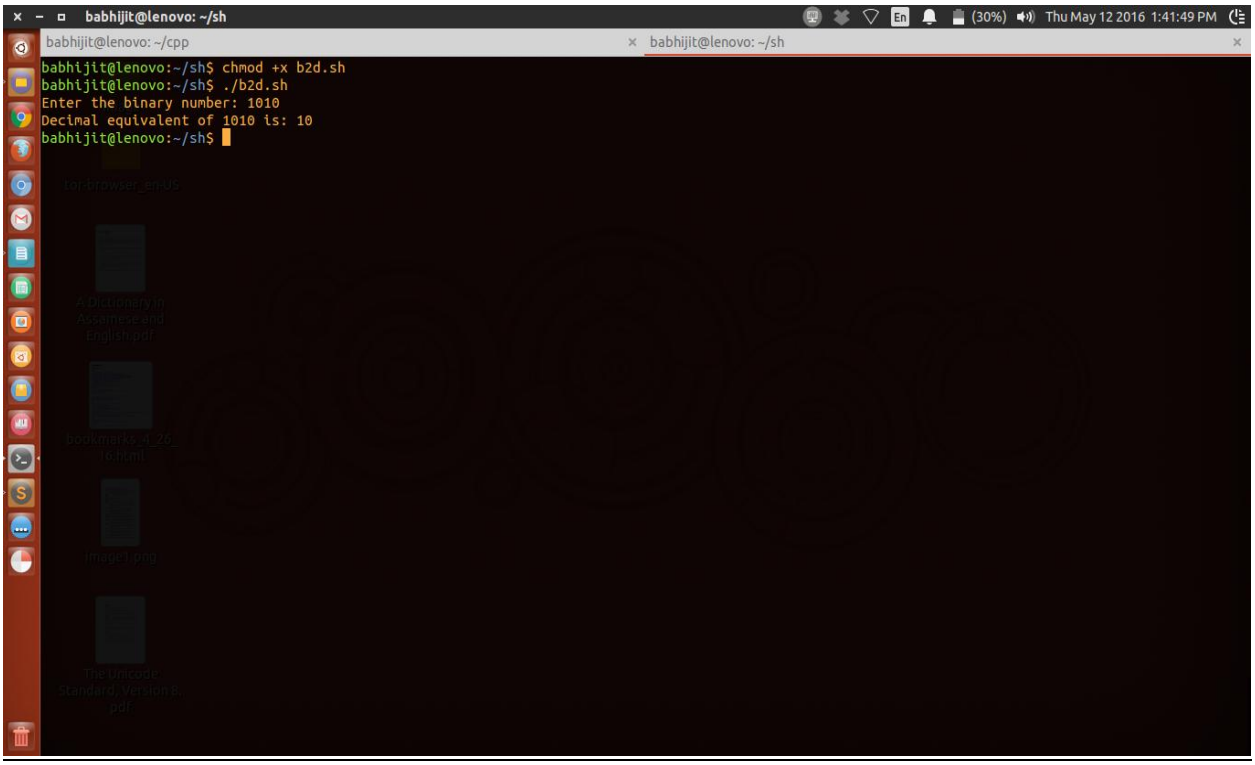
P24:- Write a shell program to convert a binary number to equivalent decimal number.

Program:-

```
#!/bin/sh
echo -n "Enter the binary number: ";
read num;
q=$num;
res=0;
k=0;
pow=1;
while [ "$q" -ne 0 ]
do
    if [ "$k" -eq 0 ]
    then
        r=$(($q%10));
        q=$(($q/10));
        k=$(($k+1));
        res=$(($res+$r \* 1));
        var=$r;
        continue;
    fi
    r=$(($q%10));
    for i in $(bash -c "echo {1..{$k}}")
    do
        pow=$(($pow \* 2));
    done
```

```
        res=$(($res+$r \* $pow));
        q=$(($q/10));
        k=$(($k+1));
    done
    echo "Decimal equivalent of $num is: $(($res))";
```

Output:-



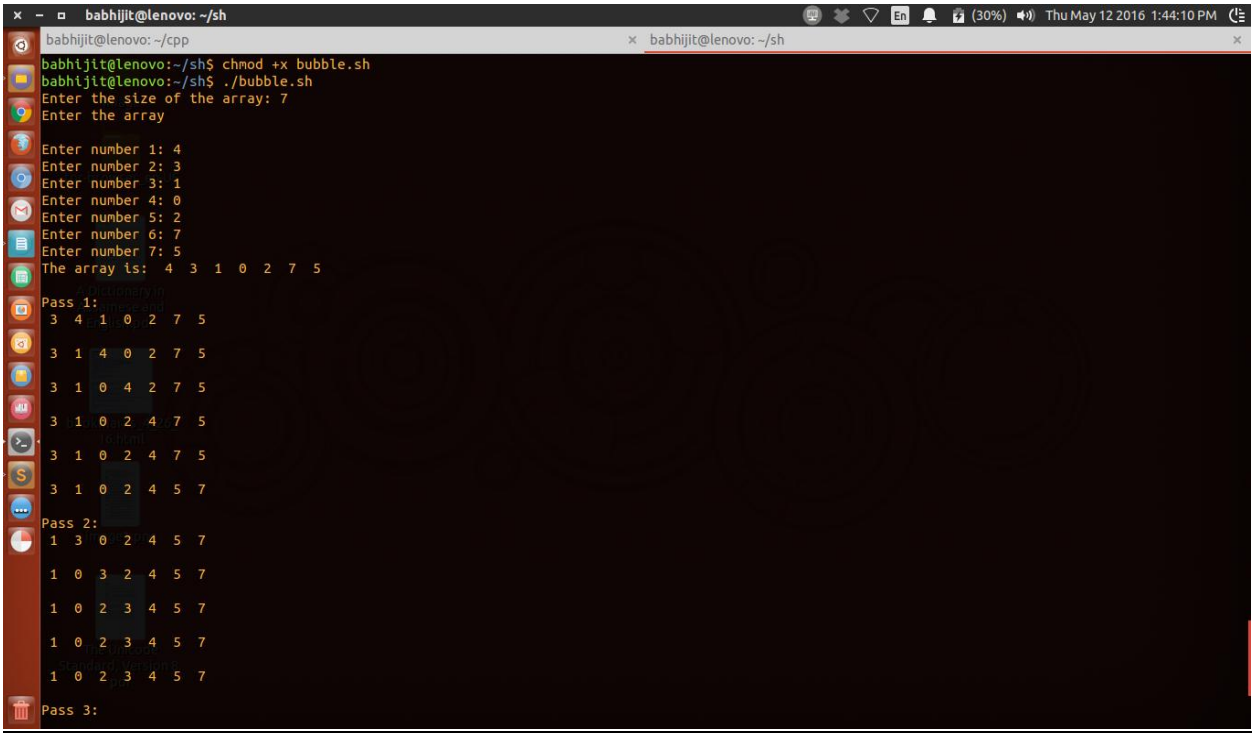
P25:- Write a shell program to implement bubble sort algorithm.

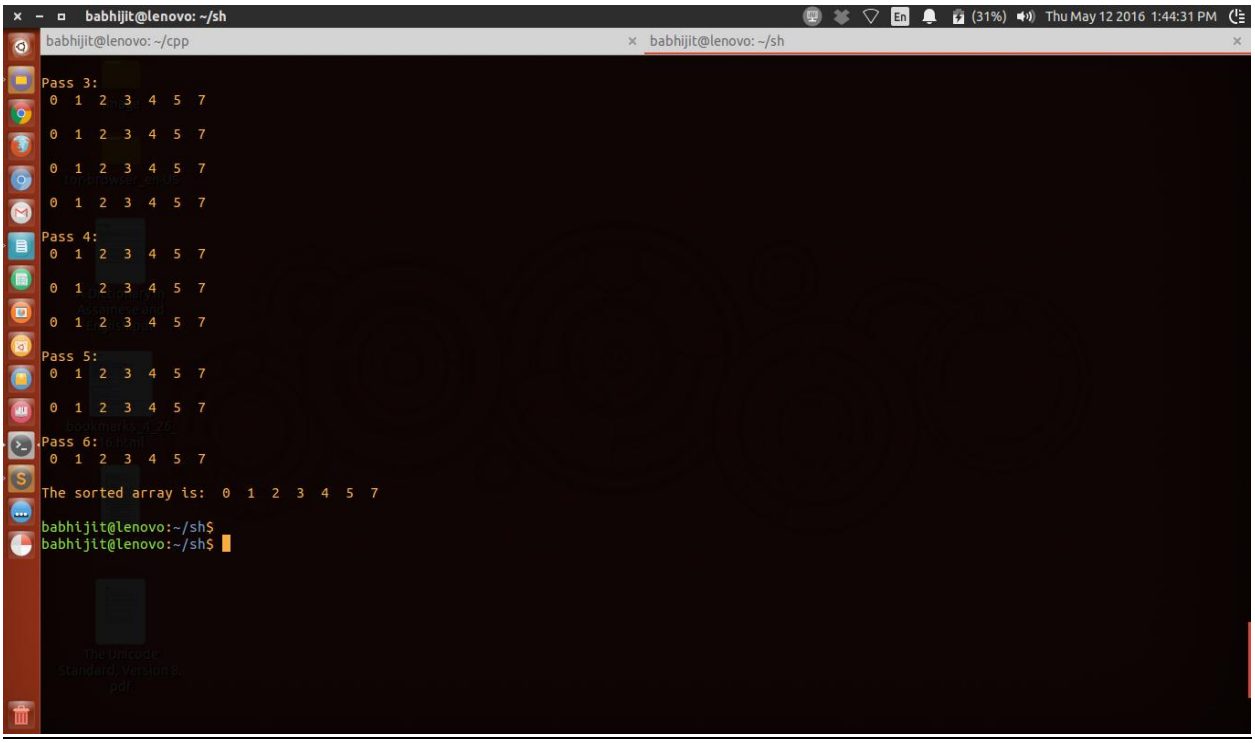
Program:-

```
#!/bin/bash
temp=0;
echo -n "Enter the size of the array: ";
read num;
echo -e "Enter the array\n";
for i in $(bash -c "echo {1..${num}}")
do
        echo -n "Enter number $i: ";
        read n;
        array[$i]=$n;
done
echo -n "The array is: "
for i in $(bash -c "echo {1..${num}}")
do
        echo -n " ${array[$i]} ";
done
echo -e "\n";
for i in $(bash -c "echo {1..$((${num}-1))}")
do
        echo "Pass $i: ";
        for j in $(bash -c "echo {1..$((${num}-$i))}")
        do
                if [ $((${array[$j]})) -gt $((${array[$j+1]})) ]
```

```
                then
                        temp=${array[$j+1]};
                        array[$j+1]=${array[$j]};
                        array[$j]=$temp;
                        #echo "true";
                fi
                for i in $(bash -c "echo {1..${num}}")
                do
                        echo -n " ${array[$i]} ";
                done
                echo -e "\n";
        done
done
echo -n "The sorted array is: "
for i in $(bash -c "echo {1..${num}}")
do
        echo -n " ${array[$i]} ";
done
echo -e "\n";
```

Output:-


babhijit@lenovo: ~/sh
babhijit@lenovo: ~/cpp
babhijit@lenovo: ~/sh
babhijit@lenovo:~/sh$ chmod +x bubble.sh
babhijit@lenovo:~/sh$ ./bubble.sh
Enter the size of the array: 7
Enter the array
Enter number 1: 4
Enter number 2: 3
Enter number 3: 1
Enter number 4: 0
Enter number 5: 2
Enter number 6: 7
Enter number 7: 5
The array is: 4 3 1 0 2 7 5
Pass 1:
3 4 1 0 2 7 5
3 1 4 0 2 7 5
3 1 0 4 2 7 5
3 1 0 2 4 7 5
3 1 0 2 4 7 5
3 1 0 2 4 5 7
Pass 2:
1 3 0 2 4 5 7
1 0 3 2 4 5 7
1 0 2 3 4 5 7
1 0 2 3 4 5 7
1 0 2 3 4 5 7
Pass 3:


babhijit@lenovo: ~/sh
babhijit@lenovo: ~/cpp
babhijit@lenovo: ~/sh
Pass 3:
0 1 2 3 4 5 7
0 1 2 3 4 5 7
0 1 2 3 4 5 7
0 1 2 3 4 5 7
Pass 4:
0 1 2 3 4 5 7
0 1 2 3 4 5 7
0 1 2 3 4 5 7
Pass 5:
0 1 2 3 4 5 7
0 1 2 3 4 5 7
Pass 6:
0 1 2 3 4 5 7
The sorted array is: 0 1 2 3 4 5 7
babhijit@lenovo:~/sh$
babhijit@lenovo:~/sh$

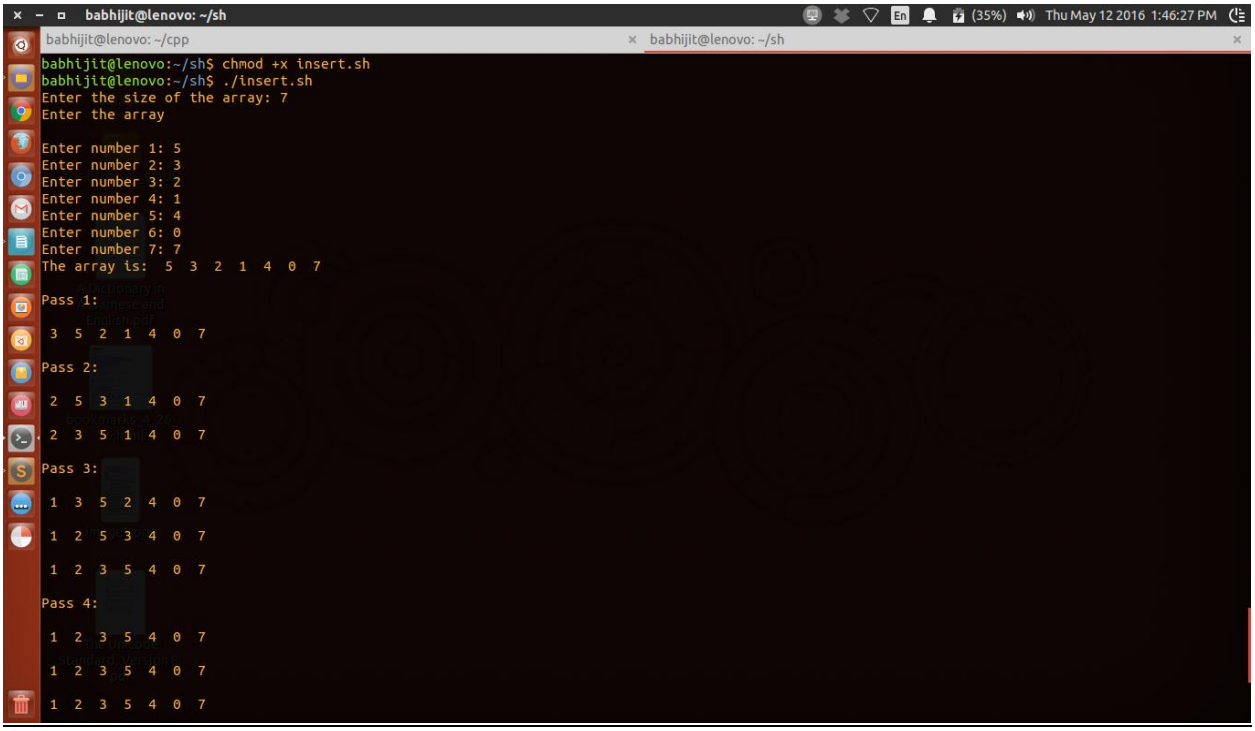
P26:- Write a shell program to implement insertion sort algorithm.

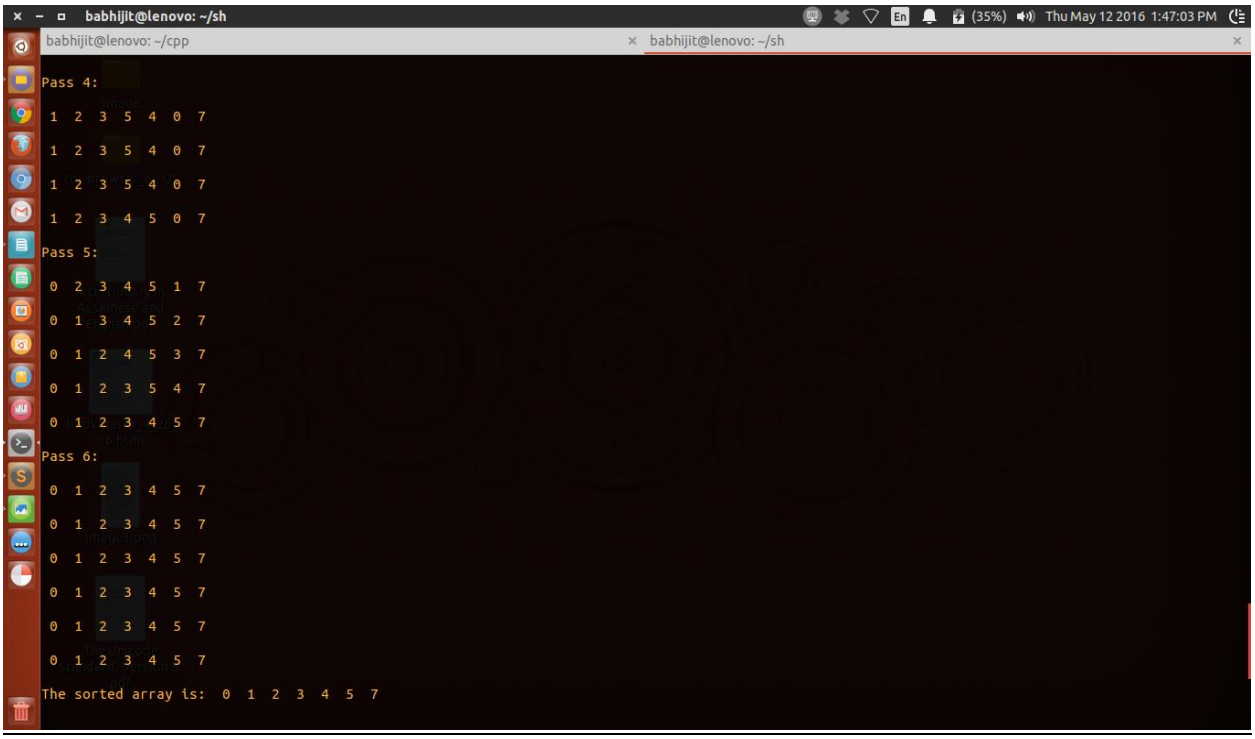
Program:-

```
#!/bin/bash
isort=();
temp=0;
echo -n "Enter the size of the array: ";
read num;
echo -e "Enter the array\n";
for i in $(bash -c "echo {1..${num}}")
do
        echo -n "Enter number $i: ";
        read n;
        isort[$i]=$n;
done
echo -n "The array is: ";
for i in $(bash -c "echo {1..${num}}")
do
        echo -n " ${isort[$i]} ";
done
echo -e "\n";
for i in $(bash -c "echo {2..$((${num}))}")
do
        echo -e "Pass $((i-1)):\n";
        for j in $(bash -c "echo {1..$((${i}-1))}")
        do
```

```
                if [ $((isort[$j])) -gt $((isort[$i])) ]
                then
                        temp=${isort[$i]};
                        isort[$i]=${isort[$j]};
                        isort[$j]=$temp;
                fi
                for k in $(bash -c "echo {1..${num}}")
                do
                        echo -n " ${isort[$k]} ";
                done
                echo -e "\n";
        done
done
echo -n "The sorted array is: ";
for i in $(bash -c "echo {1..${num}}")
do
        echo -n " ${isort[$i]} ";
done
echo -e "\n";
```

Output:-


babhijit@lenovo: ~/sh
babhijit@lenovo: ~/cpp
babhijit@lenovo: ~/sh
babhijit@lenovo:~/sh$ chmod +x insert.sh
babhijit@lenovo:~/sh$ ./insert.sh
Enter the size of the array: 7
Enter the array
Enter number 1: 5
Enter number 2: 3
Enter number 3: 2
Enter number 4: 1
Enter number 5: 4
Enter number 6: 0
Enter number 7: 7
The array is: 5 3 2 1 4 0 7
Pass 1:
3 5 2 1 4 0 7
Pass 2:
2 5 3 1 4 0 7
2 3 5 1 4 0 7
Pass 3:
1 3 5 2 4 0 7
1 2 5 3 4 0 7
1 2 3 5 4 0 7
Pass 4:
1 2 3 5 4 0 7
1 2 3 5 4 0 7
1 2 3 5 4 0 7


babhijit@lenovo: ~/sh
babhijit@lenovo: ~/cpp
babhijit@lenovo: ~/sh
Pass 4:
1 2 3 5 4 0 7
1 2 3 5 4 0 7
1 2 3 5 4 0 7
1 2 3 4 5 0 7
Pass 5:
0 2 3 4 5 1 7
0 1 3 4 5 2 7
0 1 2 4 5 3 7
0 1 2 3 5 4 7
0 1 2 3 4 5 7
Pass 6:
0 1 2 3 4 5 7
0 1 2 3 4 5 7
0 1 2 3 4 5 7
0 1 2 3 4 5 7
0 1 2 3 4 5 7
0 1 2 3 4 5 7
The sorted array is: 0 1 2 3 4 5 7

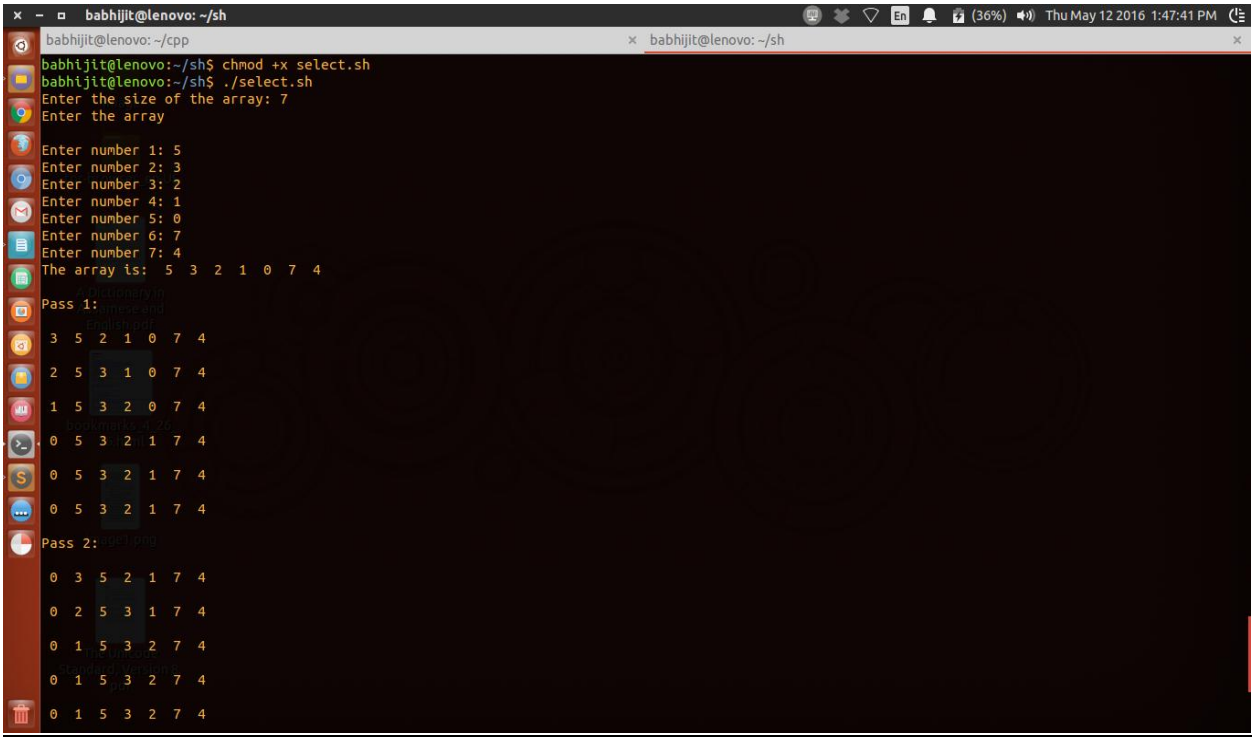
P27:- Write a shell program to implement selection sort algorithm.

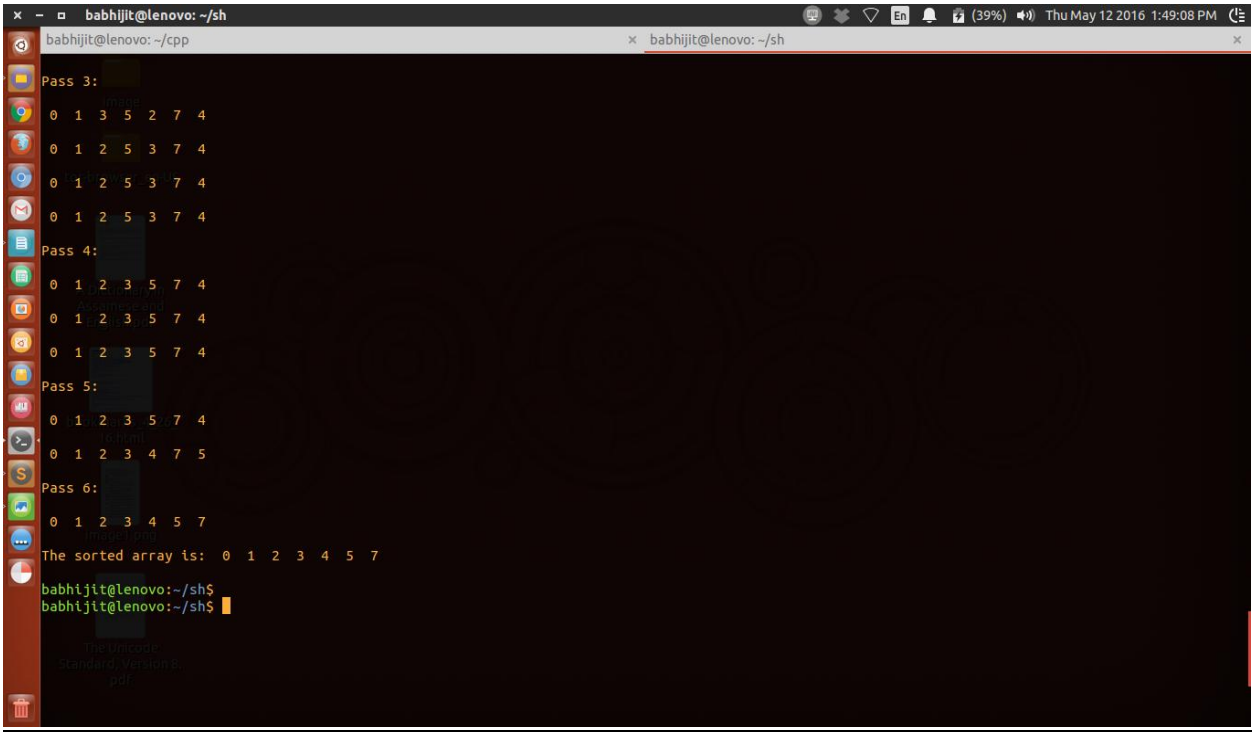
Program:-

```
#!/bin/bash
ssort=();
temp=0;
echo -n "Enter the size of the array: ";
read num;
echo -e "Enter the array\n";
for i in $(bash -c "echo {1..${num}}")
do
        echo -n "Enter number $i: ";
        read n;
        ssort[$i]=$n;
done
echo -n "The array is: ";
for i in $(bash -c "echo {1..${num}}")
do
        echo -n " ${ssort[$i]} ";
done
echo -e "\n";
for i in $(bash -c "echo {1..$((${num}-1))}")
do
        echo -e "Pass $i:\n";
        for j in $(bash -c "echo {$((${i}+1))..${num}}")
        do
```

```
                if [ $((ssort[$i])) -gt $((ssort[$j])) ]
                then
                        temp=${ssort[$i]};
                        ssort[$i]=${ssort[$j]};
                        ssort[$j]=$temp;
                fi
                for k in $(bash -c "echo {1..${num}}")
                do
                        echo -n " ${ssort[$k]} ";
                done
                echo -e "\n";
        done
done
echo -n "The sorted array is: ";
for i in $(bash -c "echo {1..${num}}")
do
        echo -n " ${ssort[$i]} ";
done
echo -e "\n";
```

Output:-


babhijit@lenovo: ~/sh
babhijit@lenovo: ~/cpp
babhijit@lenovo: ~/sh
babhijit@lenovo:~/sh$ chmod +x select.sh
babhijit@lenovo:~/sh$ ./select.sh
Enter the size of the array: 7
Enter the array
Enter number 1: 5
Enter number 2: 3
Enter number 3: 2
Enter number 4: 1
Enter number 5: 0
Enter number 6: 7
Enter number 7: 4
The array is: 5 3 2 1 0 7 4
Pass 1:
3 5 2 1 0 7 4
2 5 3 1 0 7 4
1 5 3 2 0 7 4
0 5 3 2 1 7 4
0 5 3 2 1 7 4
0 5 3 2 1 7 4
Pass 2:
0 3 5 2 1 7 4
0 2 5 3 1 7 4
0 1 5 3 2 7 4
0 1 5 3 2 7 4
0 1 5 3 2 7 4


babhijit@lenovo: ~/sh
babhijit@lenovo: ~/cpp
babhijit@lenovo: ~/sh
Pass 3:
0 1 3 5 2 7 4
0 1 2 5 3 7 4
0 1 2 5 3 7 4
0 1 2 5 3 7 4
Pass 4:
0 1 2 3 5 7 4
0 1 2 3 5 7 4
0 1 2 3 5 7 4
Pass 5:
0 1 2 3 5 7 4
0 1 2 3 4 7 5
Pass 6:
0 1 2 3 4 5 7
The sorted array is: 0 1 2 3 4 5 7
babhijit@lenovo:~/sh$
babhijit@lenovo:~/sh$