Experiment No: 08

Aim: Basic Shell Script using control structure

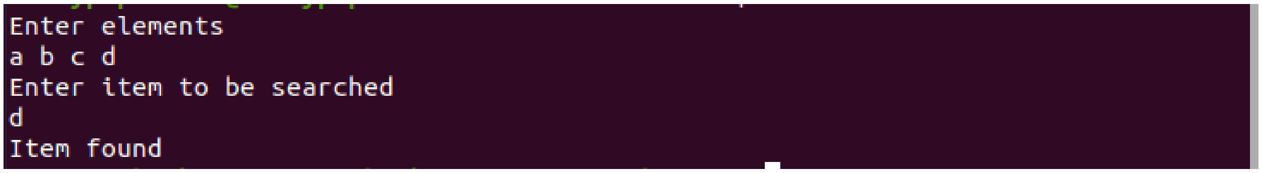
Theory:

1.Write a shell script to search whether element is present is in the list or not

Program:

```
echo   "Enter   elements"
read else

echo  "Enter  item  to  be  searched"
read item

flag=0

for e in $ele

do

if [ $e == $item ]

then

        flag=1

        break

fi

done
```

Output :



```
Enter elements
a b c d
Enter item to be searched
d
Item found
```

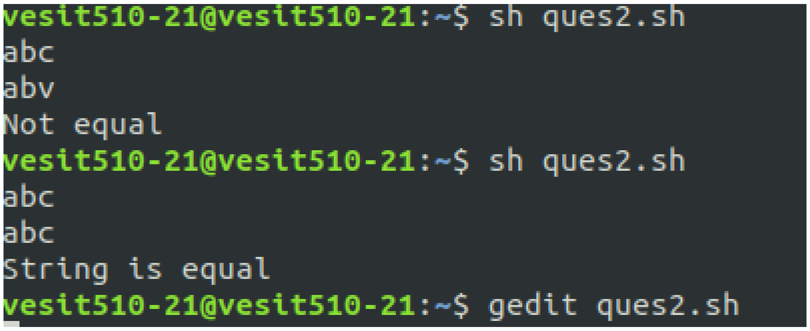
2.Write a shell script to compare two strings.

Program:

```
read   a
read b


if [ "$a" = "$b" ]
then
```

```
echo "String is equal"
else

echo "Not equal"
fi
```

Output :



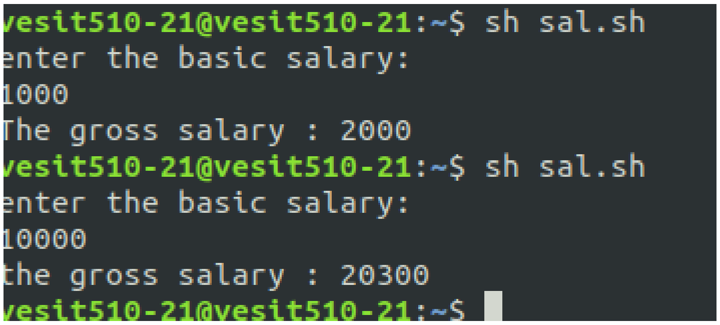
```
vesit510-21@vesit510-21:~$ sh ques2.sh
abc
abv
Not equal
vesit510-21@vesit510-21:~$ sh ques2.sh
abc
abc
String is equal
vesit510-21@vesit510-21:~$ gedit ques2.sh
```

3.Write a shell script for Employee Pay calculation

Program:

```
echo "enter the basic salary:"
read sal

if [ $sal -lt 1500 ]
then

gross=$((sal+((sal/100)*10)+(sal/100)*90))

echo "The gross salary : $gross"

fi

if [ $sal -ge 1500 ]
then

gross=$(((sal+500)+(sal/100)*98))
echo "the gross salary : $gross"

fi
```

Output :



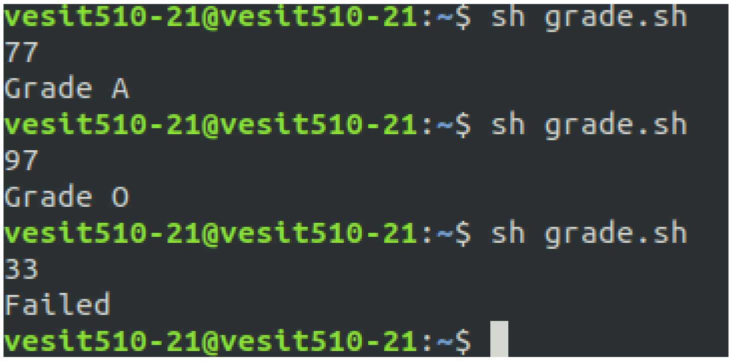
```
vesit510-21@vesit510-21:~$ sh sal.sh
enter the basic salary:
1000
The gross salary : 2000
vesit510-21@vesit510-21:~$ sh sal.sh
enter the basic salary:
10000
the gross salary : 20300
vesit510-21@vesit510-21:~$
```

4.Write a Shell script to calculate Grade

Program:

```
read grade


if [ $grade -gt 90 ]
then

echo "Grade O"

elif [ $grade -gt 75 ]
then

echo "Grade A"

elif [ $grade -gt 60 ]
then

echo "Grade B"

elif [ $grade -gt 35 ]
then

echo  "Grade  C"
else

echo   "Failed"
fi
```

<u>Output:</u>



```
vesit510-21@vesit510-21:~$ sh grade.sh
77
Grade A
vesit510-21@vesit510-21:~$ sh grade.sh
97
Grade O
vesit510-21@vesit510-21:~$ sh grade.sh
33
Failed
vesit510-21@vesit510-21:~$ 
```

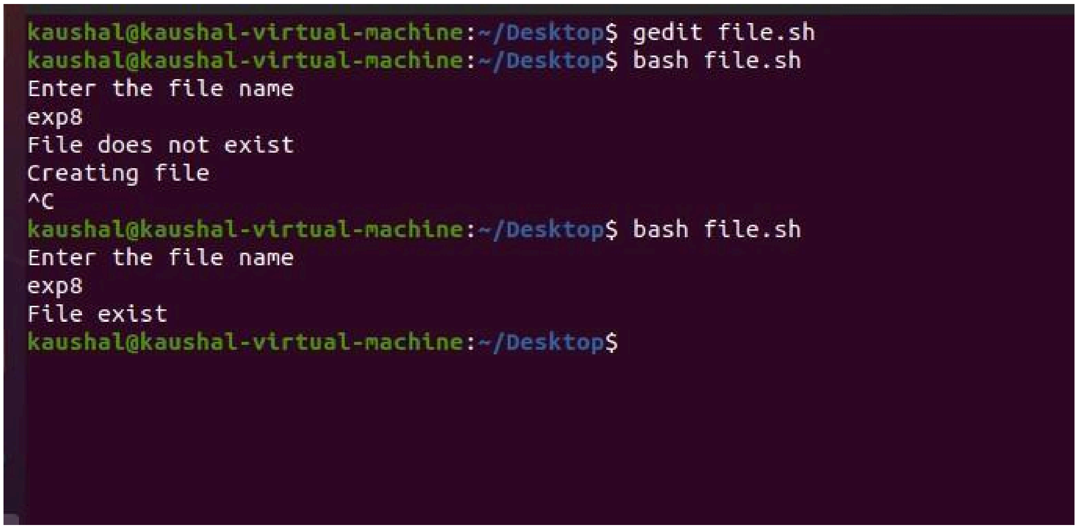
5.Write a shell script to read and check if the directory / file exists or not, if not make the directory / file.

<u>Program:</u>

```
echo "Enter the file name"

read file

if [ -e $file.txt ]

then
```

```
        echo "File exist"
else
        echo "File does not exist"
        echo "Creating file"
        cat > $file.txt
fi
```

Output:



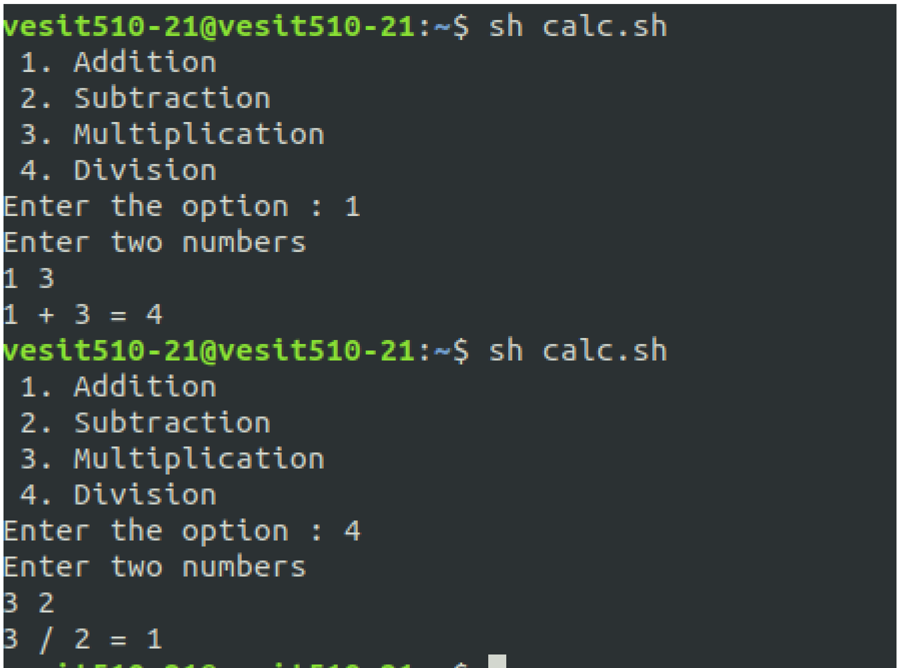
```
kaushal@kaushal-virtual-machine:~/Desktop$ gedit file.sh
kaushal@kaushal-virtual-machine:~/Desktop$ bash file.sh
Enter the file name
exp8
File does not exist
Creating file
^C
kaushal@kaushal-virtual-machine:~/Desktop$ bash file.sh
Enter the file name
exp8
File exist
kaushal@kaushal-virtual-machine:~/Desktop$
```

6.Write a shell script to implement a menu-driven calculator using a case statement.

Program:

```
echo " 1. Addition" echo
" 2. Subtraction" echo "
3. Multiplication" echo "
4. Division"

echo -n "Enter the option : "


read option

echo "Enter two numbers"
read a b

case $option in

1)      c=`expr  $a  +
$b` echo "$a + $b =
$c";;
2)      c=`expr  $a  -
$b` echo "$a - $b =
$c";;
3)      c=`expr  $a \*
$b` echo "$a * $b =
$c";;
4)      c=`expr  $a  /
$b` echo "$a / $b =
$c";;
*) echo "Invalid Option"
esac
```

Output:



```
vesit510-21@vesit510-21:~$ sh calc.sh
 1. Addition
 2. Subtraction
 3. Multiplication
 4. Division
Enter the option : 1
Enter two numbers
1 3
1 + 3 = 4
vesit510-21@vesit510-21:~$ sh calc.sh
 1. Addition
 2. Subtraction
 3. Multiplication
 4. Division
Enter the option : 4
Enter two numbers
3 2
3 / 2 = 1
vesit510-21@vesit510-21:~$
```

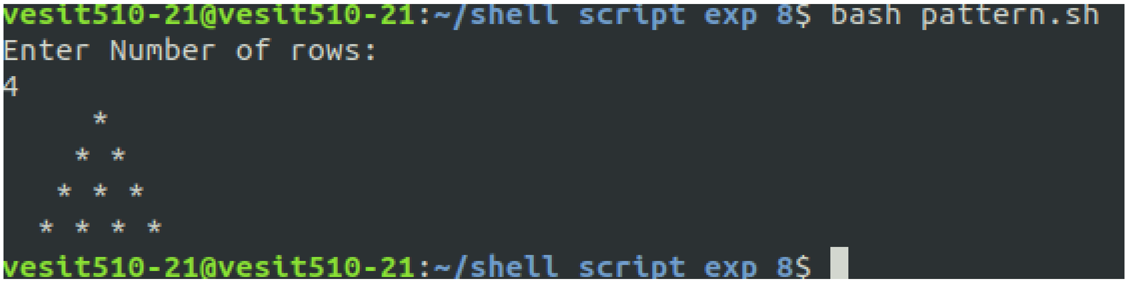
7.Write a shell script to print following pattern:

```
      *

    * *

   * * *

  * * * *
```

Program:

```
echo  "Enter  Number  of  rows:"
read rows

for  ((i=0; i<rows; i++))
do

  for((j=i;  j<=rows; j++))
  do

          echo -n " "

  done
  for((k=0;  k<=i ; k++))
  do

          echo  -n "* "
          done

          echo

  done
```

Output:



```
vesit510-21@vesit510-21:~/shell script exp 8$ bash pattern.sh
Enter Number of rows:
4
     *
    * *
   * * *
  * * * *
vesit510-21@vesit510-21:~/shell script exp 8$
```

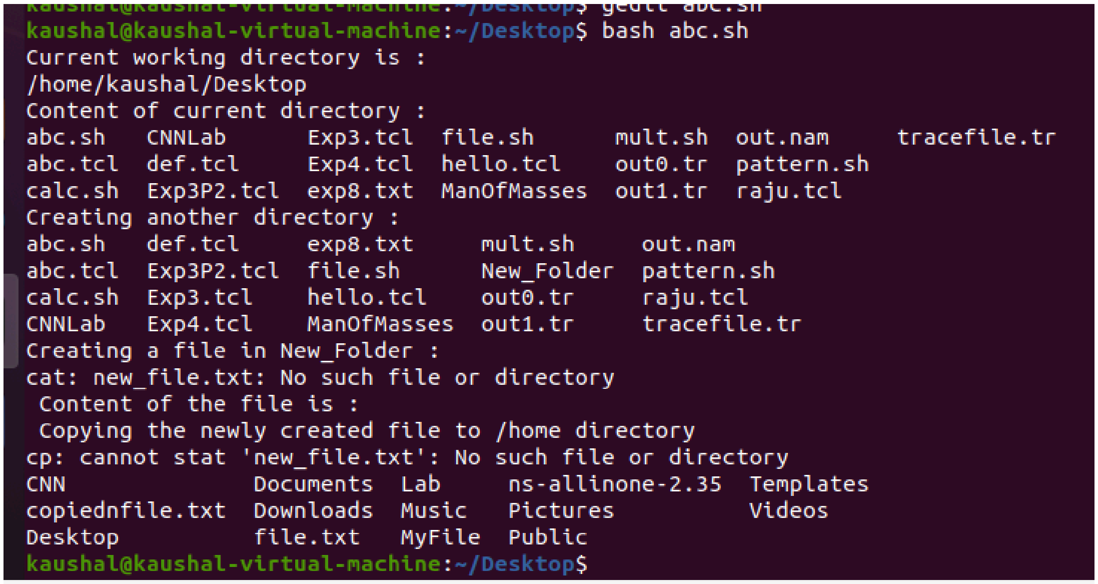
8.Write a shell script to perform operations on directory like: display name of current directory; display list of directory contents; create another directory, write contents on that and copy it to a suitable location in your home directory; etc.

Program:

```
echo "Current working directory is :"
```

```
pwd

echo "Content of current directory :" ls


echo "Creating another directory :"
mkdir New_Folder

ls


cd New_Folder

echo "Creating a file in New_Folder :" ls

gedit new_file.txt


echo " Content of the file is : `cat new_file.txt`"

echo " Copying the newly created file to /home directory" cp
new_file.txt /home

cd
ls
```

Output:



```
kaushal@kaushal-virtual-machine:~/Desktop$ gedit abc.sh
kaushal@kaushal-virtual-machine:~/Desktop$ bash abc.sh
Current working directory is :
/home/kaushal/Desktop
Content of current directory :
abc.sh   CNNLab      Exp3.tcl  file.sh      mult.sh  out.nam     tracefile.tr
abc.tcl  def.tcl     Exp4.tcl  hello.tcl    out0.tr  pattern.sh
calc.sh  Exp3P2.tcl  exp8.txt  ManOfMasses  out1.tr  raju.tcl
Creating another directory :
abc.sh   def.tcl     exp8.txt     mult.sh     out.nam
abc.tcl  Exp3P2.tcl  file.sh      New_Folder  pattern.sh
calc.sh  Exp3.tcl    hello.tcl    out0.tr     raju.tcl
CNNLab   Exp4.tcl    ManOfMasses  out1.tr     tracefile.tr
Creating a file in New_Folder :
cat: new_file.txt: No such file or directory
 Content of the file is :
 Copying the newly created file to /home directory
cp: cannot stat 'new_file.txt': No such file or directory
CNN             Documents  Lab     ns-allinone-2.35  Templates
copiednfile.txt Downloads  Music   Pictures          Videos
Desktop         file.txt   MyFile  Public
kaushal@kaushal-virtual-machine:~/Desktop$
```

9.Write a Shell script for a Menu Driven program to check if entered number is
1.Even Odd
2.Prime
3.Palindrome
4.Armstrong

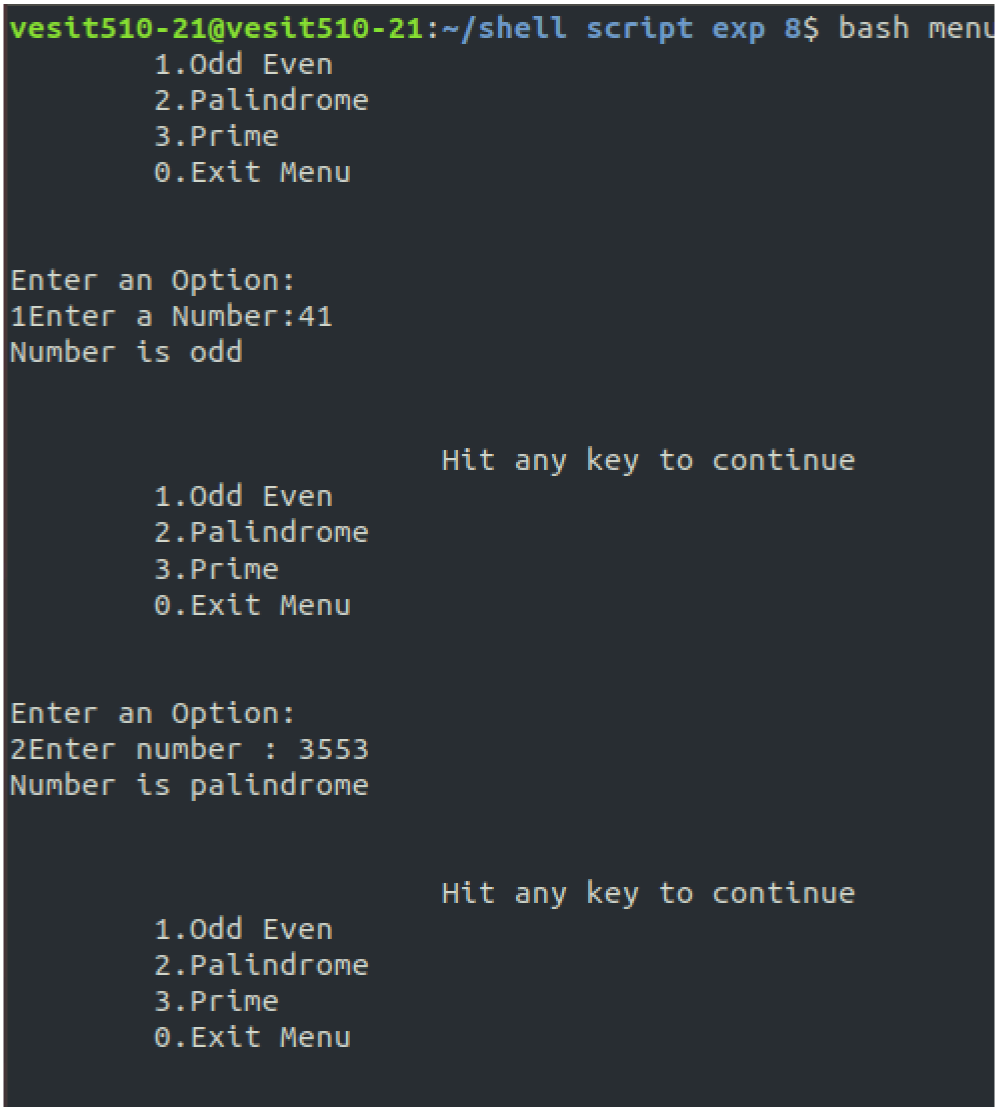
Program:

```
function prime {

echo -n "Enter the number"
read a

i=2
z=0

while [ $i -lt $a ]
do

s=`expr $a % $i`
if [ $s -eq $z ]
then

echo "Not a Prime Number"
exit

else

i=`expr $i + 1`
fi

done

echo "Prime number"

}

function oddeven {

echo -n "Enter a Number:"
read n

rem=$(( $n % 2 ))
if [ $rem -eq 0 ]
then

echo "Number is even"
else

echo "Number is odd" fi

}

function palindrome { echo
-n "Enter number : " read n

sd=0
rev=""
on=$n

while [ $n -gt 0 ]
```

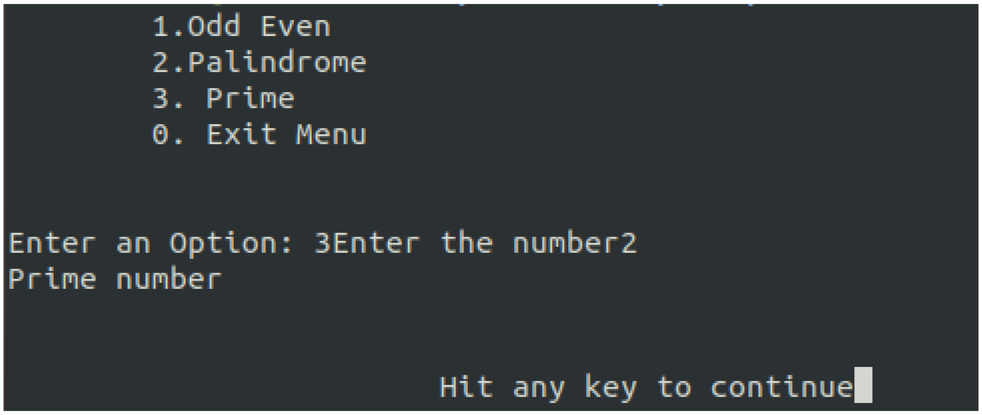
```
do
sd=$(( $n % 10 ))
```

```
 n=$(( $n / 10 ))

rev=$( echo ${rev}${sd} )
done

if [ $on -eq $rev ];
then

echo "Number is palindrome"
else

echo "Number is NOT palindrome" fi

}

function menu {

echo -e "\t1.Odd Even"
echo -e "\t2.Palindrome"
echo -e "\t3. Prime"

echo -e "\t0. Exit Menu\n\n"
echo -en "Enter an Option: "
read -n 1 option

}

while [ 1 ]
do

menu

case $option in
0)

break ;;
1)

oddeven ;;
2)

palindrome ;;
3)

prime ;;

*)
clear

echo "Sorry, wrong selection";;
esac

echo -en "\n\n\t\t\tHit any key to continue"
read -n 1 line

done
clear
```

Output:



```
vesit510-21@vesit510-21:~/shell script exp 8$ bash menu
        1.Odd Even
        2.Palindrome
        3.Prime
        0.Exit Menu


Enter an Option:
1Enter a Number:41
Number is odd



                        Hit any key to continue
        1.Odd Even
        2.Palindrome
        3.Prime
        0.Exit Menu


Enter an Option:
2Enter number : 3553
Number is palindrome



                        Hit any key to continue
        1.Odd Even
        2.Palindrome
        3.Prime
        0.Exit Menu
```



```
        1.Odd Even
        2.Palindrome
        3. Prime
        0. Exit Menu


Enter an Option: 3Enter the number2
Prime number



                        Hit any key to continue
```

Conclusion: