Q – 1. Write a Script to check whether an entered number is Palindrome or not

```
echo "Enter A Number : -"
read no
t=$no
sum=0
while [ $no -gt 0 ]
do
      d = `expr $no % 10`
      sum = `expr $sum \* 10 + $d`
      no = `expr $no / 10`
done
if[ $t -eq $sum ]
then
      echo "Enter Number Is Palindrome ";
elif
      echo "Enter Number Not Is Palindrome ";
fi
```

O/P:-

```
Enter the String : 1245421
Enter Number Is palindrome
```

Q – 2. Write a Script to check entered two string are equal or different. Also check the length of both string are greater than 10.

```
echo " Enter String 1:- "
read n1
echo " Enter String 2:- "
read n2

if test $n1 = $n2
then
       echo "Both strings are same"
else
        echo "both strings are different"
fi
```

O/P:-

```
Enter String 1:- 1246
Enter String 2:- 1246

Both strings are same
```

Q – 3 write a command to display all words in file 1 that begin with digit.

```
grep '^[0-9]' emp.txt
```

O/P:-

```
101 Hardik 12500 Surat
102 mit      12800 Surat
103 rohit    12600 Surat
104 alok     12200 Surat
105 rakesh  12700 Surat
106 jay       12900 Surat
107 savan    12300 Surat
108 nirmal   12100 Surat
109 Viral     12500 Surat
110 Vishal   12600 Surat
111 Rakesh  13000 Surat
112 Amit     11800 Surat
113 Akash   14500 Surat
114 Keyur    12500 Surat
115 Keval    12600 Surat
```

Q – 4. Write a Script to check using awk utility to create two 3*3 matrix and multiply it.

```
echo  “Enter Values To Make Matrix:-“
for I in 1 2 3
do
      for j in 1 2 3
       do
             read a[$i$j]
       done
done
for I in 1 2 3
do
      for j in 1 2 3


       done
       do
             echo –n “ $ {a[$i$j]}”
done
```

Q – 5. Write a Script to check whether the number is Armstrong or not.

```
echo "Enter A Number : -";
read no
t=$no
sum=0

while [ $no -gt 0 ]
do
	d=`expr $no % 10`
	sum=`expr $sum + $d + \* $d \* Sd`
	no=`expr $no / 10`
done

if[ $t -eq $sum ]
then
	echo "Enter Number Is Armstrong ";
elif
	echo "Enter Number Not Is Armstrong ";
fi
```

O/P:-

```
Enter The No : 153

153 is an Armstrong number"

Enter The No : 15

15 is not an Armstrong number"
```

Q – 6. Write a Script to perform the following mathematical on two inputted numbers

1). Addition

2). Subtraction

3). Multiplication

4). Division

Note: number may be integer or flout

```
echo “Enter No 1 :-”
read no1
echo “Enter No 2 :-”
read no2
echo " Press 1. For Perform Addition";
echo " Press 2. For Perform Subtraction";
echo " Press 3. For Perform Multiplication";
echo " Press 4. For Perform Division";
echo " Press 0. For Exit"
echo "Enter Your Choice :-";
read ch
case "$ch" in
        1)
                sum= `expr $no1 + $no2`;
                echo "Addition Result :- $sum" ;;
        2)
                sub= `expr $no1 - $no2`;
                echo "Subtraction Result :- $sub" ;;
        3)
                mul= `expr $no1 \* $no2`;
                echo "Multiplication Result :- $mul" ;;
        4)
                div= `expr $no1 / $no2`;
                echo "Division Result :- $div" ;;
        5) exit
esac
```

O/P:-

```
Enter No 1 :- 10
Enter No 2 :- 20
Press 1. For Perform Addition
Press 2. For Perform Subtraction
Press 3. For Perform Multiplication
Press 4. For Perform Division
Please choose a word [1,2,3 or 4] 2
Addition Result 30
```

Q – 7. Write a Script that receive strings and check both are same or different, also check the length of both strings are greater than 10.

```
echo " Enter String 1:- "
read n1
echo " Enter String 2:- "
read n2

if test $n1 = $n2
then
       echo "Both strings are same"
else
        echo "both strings are different"
fi
```

O/P:-

```
Enter String 1:- 1246
Enter String 2:- 1246
Both strings are same
```

Q – 8. Write a Script to entered 5 subject marks from command line and display percentage and appropriate class

```
 sum1=`expr $1 + $2 + $3 + $4 + $5`
```

```
echo "Sum of 5 subjects are:- " $sum1

per=`expr $sum1 / 5`
echo " Percentage:- " $per

if [ $per -ge 60 ]
then
        echo "You Got Distinction"
elif [ $per -ge 50 ]
then
        echo "You Got First class"
elif [ $per -ge  40 ]
then
        echo "You Got Second class"
else
        echo "You Are Fail"
fi
```

O/P:-

```
30 40 30 40 40
Sum of 5 Subject are : 180
```

```
Percentage : 72

You get Distinction
```

Q – 9. Write a Script to check entered character is uppercase, lowercase, numeric, digits or special character.

```
char=""

echo -n "Enter a one character : "
```

```
read char
if [ -z $(echo $char | sed -e 's/[0-9]//g') ]
then
       echo "$char is Number/digit"
elif [ -z $(echo $char | sed  -e 's/[A-Z]//g') ]
then
        echo "$char is UPPER character"

elif [ -z $(echo $char | sed -e 's/[a-z]//g') ]
then
        echo "$char is lower character"
else
       echo "$char is Special symbol"
fi
```

O/P:-

```
Enter a one character: @
@ is Special character
```

Q – 10. Write a Script that accept filenames and a number N from command-line and display last n_lines of each file. (do not use tail command)

```
n=$1
shift
files=$*
```

```
for file in $files
do
        total=`grep -c ".*" $file`

        n=`expr $total - $n`
        n=`expr $n + 1`

        echo "\n\n\t\t == Content of file \"$file\" ==\n\n"
        while [ $n -le $total ]
        do
                head -$n $file|sed -n '$p'
                n=`expr $n + 1`
        done
done
```

O/P:-

```
Emp.txt 5
```

Q – 11. Write an awk script to display file contents in reverse. (I.e. last-line should be display first… first-line should be display last)

```
echo "Enter File Name:\c"
read File
cat  $File | awk `{ print Strrev <$0> } | $p
```

Q – 12. Write a Shell Script for the following

- Consider the stud.dat data file having fields: stud_id, name, city, and state. Each file separated by |. Write a menu driven script to perform following task.

  1).Append record to a data file

  2).Delete a record form a file

  3).Display record city-wise

```
echo "\n\tWelcome to student management system\n\n"
echo "1 For Enter new record"
echo "2 For Delete record"
echo "3 For Display record City-wise"
echo "Which one ? :\c"
read ch

case $ch in
1)
      echo "Enter Student Roll No : \c"
      read rno
       echo "Enter Student Name : \c"
       read name
       echo "Enter City : \c"
       read city
       echo "Enter State : \c"
```

```
        read state
        echo "$rno|$name|$city|$state">>stud.dat
        if [ $? ]
        then
                echo "Record added successfully"
        else
                 echo "Record coudn't added"
        fi
        ;;
2)
        echo "Enter Student Roll No to be deleted : \c"
        read rno
        grep -v "^$rno" stud.dat > tmp
        rm stud.dat
        mv tmp stud.dat
        echo "Record Deleted"
        ;;
3)
        awk -F"|" '
        BEGIN {
                printf "Enter City to be searched : "
                 getline city < "/dev/tty11"
                 printf "\nSearching ... \n\n"
                 flag=0
        }
         {
                if($3==city)
                 {
                        printf "%s\n",$0
                           flag=1
                 }
         }
         END {
                if(flag==0)
                        printf "\n Searching Complete no match found \n"
                 else
                          printf "\nMatch found records are listed above\n\n"
         }' stud.dat
         ;;
*)
        echo "Wrong Selection"
esac

echo "Wanna Continue ? (Y/N) \c"
read ch

case $ch in
        [(y|Y)]*)
```

```
              sh scriptname
              ;;
       [(n|N)]*)
              echo "\n\tThanks for Visiting\n"
              ;;
       *)
              sh scriptname
esac
```

O/P:-

```
Welcome to student management system
1 for Enter new record
2 for Delete record
3 for Display record City-wise
Which one ? :

              Enter Student Roll No :
              Enter Student Name :
              Enter City :
              Enter State :
              Record added successfully

              Enter Student Roll No to be deleted :
```

```
Record Deleted

Enter City to be searched :
 Searching ...
Searching Complete no match found
Match found records are listed above

Thanks for Visiting
```

Q – 13. Write a Script that receives any number of filenames as argument and then count number of consonants, vowels, digits and special characters in each file.

```
file=$1
v=0

if [ $# -ne 1 ]
then
        echo "$0 fileName"
        exit 1
fi
if [ ! -f $file ]
then
        echo "$file not a file"
        exit 2
fi
while read -n 1 c
do
        l=$(echo $c | tr [:upper:] [:lower:])
        [[ "$l" == "a" || "$l" == "e" || "$l" == "i" || "$l" == "o" ||
"$l" == "u" ]] && (( v++ ))
done < $file

echo "Vowels : $v"
echo "Characters : $(cat $file | wc -c)"
echo "Blank lines : $(grep  -c '^$' $file)"
echo "Lines : $(cat $file|wc -l )"
```

O/P:-

```
Vowels :
Characters :
Blank lines :
Lines :
```

Q – 14. Write a Script using awk utility to create two 3*3 matrix and multiply it.

```
echo  "Enter Value To Make Matrix:-"
for I in 1 2 3
do
      for j in 1 2 3
       do
             read a[$i$j]
       done
done
for I in 1 2 3
do
      for j in 1 2 3
       done


       do
       echo –n "$ {a[$i$j]}"
done
```

Q – 15. Write a Script that merges two file alternatively in reverse (i.e. last lines of two files, second last line of two files, so on).

```
echo Enter first filename

read first

echo Enter second filename

read second

cat $first > third

cat $second >> third

echo After concatination of contents of entered two files

echo ---------------------------------------------------

cat third | more

echo ---------------------------------------------------
```

O/P :-

```
Enter first filename :- Hardik Nangah

Enter second filename :- Shailesh Koladiya

After concatination of contents of entered two files

 ---------------------------------------------------

 Hardik Nangah  Shailesh Koladiya

 ---------------------------------------------------
```

Q – 16. Write a Script that removes only empty files from current directory.

```
file_names=`find . -name "*.*" -size 0c`
for i in $file_names
do
        print "Do you want to delete the file?"
read input

        if ans='y' or 'Y'
        then
                rm -f $i
        else
                #do nothing
                echo "File not deleted"
        fi
done done
```

O/P:-

```
Mytxt.txt
Do you want to delete the file? Y
```

Q – 17. Write an awk Script to prints the file myfile.txt. The output should be such that there should be only 25 characters in each line. If a line in the file exceeds 15 characters, the reaming characters should be printed in the next line.

```
if [ $# -eq 0 ]
then
   echo "$0:Error command arguments missing!"
   echo "Usage: $0 start_line   uptoline   filename"
   echo "Where start_line is line number from which you would
like to print file"
   echo "uptoline is line number upto which would like to print"
   echo "For eg. $0 25 25 myfile"
   echo "Here from myfile total 25 lines printed starting from line
no. 25 to"
   echo "line no 25."
   exit 1
fi

if [ $# -eq 25 ]
then
       if [ -e $3 ]
       then
              tail +$1 $3 | head -n$2
       else
              echo "$0: Error opening file $3"
```

```
                exit 2
        fi
else
         echo "Missing arguments!"
fi
```

Q – 18. Write an awk Script that calculates frequency of palindrome words in a text file.

```
echo "Enter the String :\c"
read S
r = ` echo $c | rev `
if [$s = $r]
then
      echo  | awk   -F  "|"  ` ~ /$s/ "| wc "l  < $s
fi
```

O/P:-

```
Enter the String : Rakesh
5
```

Q – 19. Develop a shell script that creates 100 files with the same name bca0001 up to bca100

```
clear
read  n
while [  $n  -le 100 ]
do
       cat > bca[$n]
        n=`expr $n + 1`
done
```

Q – 20. The distance between two cities (in terms of kms) in input through the keyboard. Write a shell script to convert it in meters, feet’s inches and cms.

```
echo "Enter Distance In Kilometers"
read dist
meter="expr $dist \* 1000"
centi="expr $meter \* 100"
inches="expr $centi / 2.54 | bc"
feet="expr $inches / 12 | bc"

echo "Distance in meters = " $meter
echo "Distance in Centimeters =" $centi
echo "Distance in inches = " $inches
echo "Distance in feet = " $feet
```

O/p:-

```
Enter the distance in kilometers 5
Distance in meters = 5000
Distance in Centimeters =500000
Distance in inches =196850.39
Distance in feet = 16404.2
```

Q – 21. Write a Shell Script that output number of words that begins with

- Capital alphabet following by filename for file given as command line argument. Necessary validation is expected.

OR

- Write a Shell Script that outputs filenames followed by granted permissions of only those file having same permission for both owners as well as group.

```
echo “Enter File Name: \c”
read File
ls  [a-zA-Z]  *.* <$File ;
```

```
echo “Enter File 1 Name :\c”
read File1
echo “Enter File2 Name :\c”
read File2

chmod   u+rwx ,g+rwx ,o-rwx <$File1 | cmp $File2
chmod   u+rwx ,g+rwx ,o-rwx <$File2 |cmp $File1
```

Q – 22. Write a Script to perform mathematical operations using menu.

```
echo “Enter No 1 :-”
```

```
read no1
echo “Enter No 2 :-”
read no2

echo " Press 1. For Perform Addition";
echo " Press 2. For Perform Subtraction";
echo " Press 3. For Perform Multiplication";
echo " Press 4. For Perform Division";
echo " Press 0. For Exit"

echo "Enter Your Choice :-";
read ch

case "$ch" in
        1)
                sum= `expr $no1 + $no2`;
                echo "Addition Result :- $sum" ;;
        2)
                sub= `expr $no1 - $no2`;
                echo "Subtraction Result :- $sub" ;;
        3)
                mul= `expr $no1 \* $no2`;
                echo "Multiplication Result :- $mul" ;;
        4)
```

```
            div= `expr $no1 / $no2`;
            echo "Division Result :- $div" ;;
        5) exit
esac
```

O/P:-

```
Enter No 1 :- 10
Enter No 2 :- 20
Press 1. For Perform Addition
Press 2. For Perform Subtraction
Press 3. For Perform Multiplication
Press 4. For Perform Division
Please choose a word [1,2,3 or 4] 2
Addition Result 30
```

Q – 23. Write a Shell Script to check whether the given file is empty or not.

```
echo "Enter Youe File Name :- ";
read file
```

```
if test -z file
then
	echo "File Is Empty ";
else
	echo "File Is Not Empty";
fi
```

O/P:-

```
Enter Your File Name :-
Empp.txt
File Is Empty

Enter Your File Name :-
emp.txt
File Is Not Empty
```

Q – 24. Write a Shell Script that will take a file name and a positives integer number from a command line. Display content of each line of a file maximum to a number given on command line. Appropriate data validation is expected.

Ans:

Vi 24.sh

```
#sed -n "1,$2p" $1;

#        or

head -n $2 $1;
```

O/P:-

```
Sh 24.sh emp.txt 7
101 Hardik 12500 Surat
102 mit 12800 Surat
103 rohit 12600 Surat
104 alok 12200 Surat
105 rakesh 12700 Surat
106 jay 12900 Surat
107 savan 12300 Surat
```

Q – 25. Write a Shell Script to list regular file that consist of exactly five words in at least one of the line.

```
echo “File Name With Word Which Have 5 Word :”
grep “^…..$” emp.txt
```

O/P:-

```
Emp.txt :surat


Emp.txt :Dahod


sal.txt :Daman
```