1. Create a file text.txt, copy text.txt to sample.txt, count the number of lines ,words and characters, and sort the file and display the original and sorted file.?

Ans:

Create a file text.txt

```
cat >text.txt
This is a sample text file
```

Copy files to text.txt to sample.txt

```
cp text.txt sample.txt
```

number of lines

```
echo "Number of lines = " && wc -l sample.txt
Output
Number of lines = 1 sample.txt
```

number of words

```
echo "number of words = "&& wc -w sample.txt
Output
number of words = 6 sample.txt
```

number of characters

```
echo "number of characters = "&& wc -m sample.txt
Output
number of characters = 27 sample.txt
```

Sort the file

```
sort text.txt sort.txt
```

display the original

```
cat text.txt
Output
This is a sample text file
```

display the sorted file

```
cat sort.txt
Output
This is a sample text file
```

2. Illustrate the working of basic Linux commands to

a. Check the present working directory

b. Display current path settings
c. Include current working directory in path settings
d. List the contents of a directory using wild cards *, ?
e. To create and delete multiple sub directories
f. Create a directory hierarchy “bca/exam/internal”
g. Change primary prompt to current date

Ans)

a)

pwd

Output

/home/arjunsanthosh

b)

echo $PATH

Output

/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/snap/bin

c)

export PATH=$PATH:/home/arjunsanthosh

d)

ls *

Output

sample.txt sort.txt text.txt

Desktop:

Documents:

Downloads:

'lab cycle 2[1].docx' 'org.gimp.GIMP(1).flatpakref' org.gimp.GIMP.flatpakref record yellow-sport-car-with- black-autotuning-bridge.jpg

Music:

Pictures:

Public:

snap:

firefox snapd-desktop-integration

Templates:

Videos:

e)

mkdir dir1 dir2 dir3

rm -r dir1 dir2 dir3

f)

mkdir -p bca/exam/internal

g)

export PS1="\d$ "

Output

Sat Jul 06$

3. Command to change all lower case letters in a file to upper case letters?

Ans)
tr '[:lower:]' '[:upper:]' <text.txt> output.txt
Output
cat output.txt
THIS IS A SAMPLE TEXT FILE

4. Create file student which contains no, name, and mark. Display the total number of students, sort the file based on mark and show the student who got highest mark

Ans)
create a file
cat >student.txt
1 Arjun 100
2 Arun 90
3 Gopal 98
4 Abel 89
5 Nithin 89

display total number of students
echo "Total number of students : "&& wc -l <student.txt
Output
Total number of students : 5

sort the file based on mark
sort -k3 -n student.txt > mark.txt
show the student who got highest mark

echo "student who got the higest mark is : "&& sort -k3 -nr student.txt | head -n 1
Output
student who got the higest mark is : 1 Arjun 100

5.Replace all characters in a file to # except vowels and digits.

Ans)
cat >text1.txt
hai this is a sample file 1234

echo "Change all characters to # except vowels and digits:" && sed 's/[^AEIOUaeiou0-9]/#/g' text1.txt
Output
Change all characters to # except vowels and digits:
#ai###i##i##a##a###e##i#e#1234

5.Translate each word in the input file to separate line

Ans)

```
echo "translating all the word in to new line :" && sed 's/ /\n/g' text1.txt
translating all the word in to new line :
hai
this
is
a
sample
file
1234
```

SHELL PROGRAMS

1 .Program to accept the name of the file from the standard input and then performs the following operations: enter 5 values in a file, sort the file, and list unsorted and sorted file.

Ans)

```
#!/bin/bash

        echo "Enter the name of the file:"
        read filename

        touch $filename

        echo "Enter 5 values (one per line, press Enter after each):"
        for (( i=1; i<=5; i++ ))
        do
                read value
                echo $value >> $filename
        done

        echo "Unsorted file contents:"
        cat $filename

        sort -o "$filename" "$filename"

        echo "Sorted file contents:"
        cat $filename
```

Output

```
bash qs1.sh
Enter the name of the file:
```

Arjun10
Enter 5 values (one per line, press Enter after each):
z
r
a
d
e
Unsorted file contents:
z
r
a
d
e
Sorted file contents:
a
d
e
r
z

2 .Program to read a student register number ,name ,and four subject's marks and print whether he is passed or fail?

Ans)

```
#!/bin/bash

echo "Enter the student name:"
read name

echo "Enter the student Register number:"
read reg_no

echo "Enter the four subjects marks one by one (out of 100):"
read m1 m2 m3 m4

if [ $m1 -ge 40 ] && [ $m2 -ge 40 ] && [ $m3 -ge 40 ] && [ $m4          -ge 40 ];
then
        total=$(expr $m1 + $m2 + $m3 + $m4)
        echo "Total marks: $total"
        echo "Student Passed"
else
        echo "Student Failed"
fi
```

Output
Enter the student name:
Arjun

Enter the student Register number:
101
Enter the four subjects marks one by one (out of 100):
100 100 98 99
Total marks: 397
Student Passed

3 .Program to check whether two strings are equal or not, length is 0 or not and concatenating two strings?

Ans)

```
#!/bin/bash
echo "Enter the first string:"
read string1
echo "Enter the second string:"
read string2
echo "Checking if the strings are equal or not, if their lengths are zero or not, and concatenating the two strings."
if [ "$string1" = "$string2" ];
then
        echo "Strings are equal"
else
        echo "Strings are not equal"
fi

if [ -z "$string1" ]
then
        echo "Length of the first string is zero"
else
        echo "Length of the first string is not zero"
fi

if [ -z "$string2" ]
then
        echo "Length of the second string is zero"
else
        echo "Length of the second string is not zero"
fi
concatenated_string="$string1$string2"
echo "Concatenating two strings: $concatenated_string"
```

Output
Enter the first string:
hai everyone
Enter the second string:
have a nice day

Checking if the strings are equal or not, if their lengths are zero or not, and concatenating the two strings.

Strings are not equal
Length of the first string is not zero
Length of the second string is not zero
Concatenating two strings: hai everyonehave a nice day

4 .Program to print the first n Fibonacci series?

Ans)

```
#!/bin/bash
echo "Enter the number of Fibonacci numbers to print:"
read n
a=0
b=1
echo "Fibonacci series for first $n numbers:"
for (( i=0; i<n; i++ ))
do
        echo -n "$a "
        next=$((a + b))
        a=$b
        b=$next
done
```

Output

Enter the number of Fibonacci numbers to print:4
Fibonacci series for first 4 numbers:
0 1 1 2

5 .Program to check whether two strings are equal or not, length is 0 or not and concatenating two strings?

Ans)

```
#!/bin/bash

factorial() {
if [ $1 -eq 0 ] || [ $1 -eq 1 ]; then
echo 1
else
echo $(( $1 * $(factorial $(( $1 - 1 ))) ))
fi
}

echo "Enter the value of n:"
read n

echo "Enter the value of r:"
read r
```

```
if [ $r -gt $n ]
then
echo "Error: r should be less than or equal to n."
elif [ $n -ge 0 ] && [ $r -ge 0 ]
then
nCr=$(( $(factorial $n) / ( $(factorial $r) * $(factorial $ (( $n - $r ))) ) ))
    echo "The value of $n C $r is: $nCr"
else
echo "Error: n and r should be non-negative integers."
fi
```

Output
Enter the value of n:
4
Enter the value of r:
3
The value of 4 C 3 is: 4

6 .Program to find the sum of elements in a array?

Ans)

```
#!/bin/bash
declare -a ar
echo "Enter the number of elements in the array:"
read n
echo "Enter the elements:"
sum=0
for ((i=0; i<n; i++))
do
    read ar[i]
    sum=$((sum + ar[i]))
done

echo "Array elements are: ${ar[*]}"
echo "Sum of the array is: $sum"
```

Output
Enter the number of elements in the array:
3
Enter the elements:
1
2
1
Array elements are: 1 2 1
Sum of the array is: 4

7 .Write shell program to get a subdirectory name from user and list the contents inside the directory. Also display how many entries of the subdirectory start with file name “ab” .

Ans)

```
#!/bin/bash
echo "Enter the name of the subdirectory:"
read subdir
if [ -d "$subdir" ]
then
        echo "Contents of the directory $subdir:"
        ls "$subdir"
        count=$(ls "$subdir" | grep -c '^ab')
        echo "Number of entries starting with 'ab': $count"
else
        echo "Error: $subdir is not a valid directory."
fi
```

Output

```
Enter the name of the subdirectory:
Downloads
Contents of the directory Downloads:
'lab cycle 2[1](1).docx'  'lab cycle 2[1].docx'           'org.gimp.GIMP(1).flatpakref'
org.gimp.GIMP.flatpakref      record   yellow-sport-car-with-black-autotuning-bridge.jpg
Number of entries starting with 'ab': 0
```

8 .Write a Shell program to get two file names from the user and check whether they are same or not. If both the files are same delete the second one.

Ans)

```
#!/bin/bash
echo "Enter the name of the first file:"
read file1
echo "Enter the name of the second file:"
read file2
if [ -e "$file1" ] && [ -e "$file2" ]
then
  if  cmp -s "$file1" "$file2"
    then
echo "The files are the same."
rm "$file2"
echo "The second file '$file2' has been deleted."
  else
echo "The files are different."
  fi
else
        echo "One or both of the files do not exist."
fi
```

Output
Enter the name of the first file:
file.txt
Enter the name of the second file:
file2.txt
The files are the same.
The second file 'file2.txt' has been deleted.

9 .Write a shell script to get three file names and a directory name as command line argument and create the three files and the directory in in the current working directory. Display appropriate message if command line arguments are less than four.

Ans)

```
#!/bin/bash
if [ $# -lt 4 ]
then
        echo "Error: You need to provide exactly three file names and one directory name."
        echo "Usage: $0 file1 file2 file3 directory"
        exit 1
fi
file1=$1
file2=$2
file3=$3
dir=$4
touch "$file1"
touch "$file2"
touch "$file3"
mkdir -p "$dir"
echo "Created files: $file1, $file2, $file3"
echo "Created directory: $dir"
```