1.1 Write a shell program to receive a specific word and file name as command line arguments and perform the options ,
Check whether the file is empty or not, If not empty then,

a. Display the no of lines that contains only the specific word

```
file=$1
if [ -e "$file”]
then
echo "Enter word:"
read word
echo -n "Number of lines:"
grep -c "$word" $file
else
echo "File not Found!!"
fi
```

b. Delete the lines of the file contains the specific word.

```
file=$1
echo "Enter Word"
read word
echo "After Deletion:"
grep -v "$word" $file>temp
cat temp
```

1.2 Write a shell script to print the sum of first n odd numbers.

```
echo "Enter n:"
read n
sum=0
for((i=0;i<=n;i++))
do
if((i%2!=0))
then
sum=$((sum+i))
fi
done
echo "Sum is:$sum"
```

2.1 Write a shell script that receives any number of file & directory names as arguments checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file, then check the file contains odd number of lines or not.

```
file=$1
if [ -f $file ]
then
echo "its a file!"
nl=$(grep -c "" $file)
if((nl%2!=0))
then
echo "it contains odd number of lines"
else
echo "it contains even number of lines"
fi
else
echo "its a directory!"
          fi
```

2.2 Write a shell script to check whether the given number is palindrome or not.

```
echo Enter the string
read s
echo $s>temp
rvs="$(rev temp)"
if [ $s = $rvs ]
then
echo "it is palindrome"
else
echo " it is not a Palindrome"
fi
```

3.1 Write a shell script that receives a file name as argument checks if every argument supplied is an ordinary file then display the first words of the lines from 4th to 8th.

```
echo "enter the file"
read file
if [ -e $file ]
then
if [ -f $file ]
then
        echo "ordinary file"
fi

if [ -d $file ]
then
        echo "directory file"
fi
else
        echo "file not exit"
        fi
        frt=4
        sec=8
        len=`expr $sec - $frt`
        len=`expr $len - 3`
        sec=`expr $sec - 3`
        head -$sec $file | tail -$len
```

3.2 Write a shell script to find the sum of digits of a given number.

```
echo "Enter a number"
read num

while [ $num -gt 0 ]
do
  let mod=$num%10
  let sum=$sum+$mod
  let num=$num/10

done

echo "Sum of digits: $sum"
```

4.1 Write a shell script that receives any number of file & directory names as arguments checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a directory then print the total number of files within that directory.

```
echo "enter the file"
read file
if [ -d $file ]
then
echo "directory file"
echo -n "Number of files:"
ls -1 | wc -l
fi
else
echo "file not exit"
        fi
```

4.2 Write a shell script to print the following pattern

```
*

*      *

*     *        *

*      *        *       *
```

```
for((i=0;i<3;i++))

do

for((j=0;j<=i;j++))

do

echo -n "*"

done

echo ""

done
```

5.1 Write a shell script that receives a file or directory name as argument check whether the supplied name is a directory. If is a directory then print all the file details in that directory in reverse order.

```
echo "enter the file"

read file

if [ -d $file ]

then

echo "directory file"

echo -n "Number of files:"

ls -r

fi

else

echo "file not exit"

fi
```

5.2 Write a shell script to print Fibonacci series up to limit n

```
echo "Enter limit:"
read n
a=0
b=1
for((i=0;i<n;i++))
do
echo $a
c=$((a+b))
a=$b
b=$c

done
```

6.1 Write a menu- driven program for the following options

a) Accept a file name, starting and ending line numbers as arguments and display all the lines between the given line numbers from the file.

b) Check whether the file has write permission or not, if there is no write permission give write permission to that file

```
echo "enter your choice"
echo "1"
echo "2"
read ch
echo " enter your file "
read file1
if((ch==1))
then
echo " read start "
read start
echo " read end "
read end
end=$((end-1))
head -$end $file1 | tail -$((end-start))
else
if [ -w $file1 ]
then
 echo " it has write permission "

else

 echo " it has no permission "
chmod a+w $file1
fi
fi
```

6.2 Write a shell program that takes a number as command line argument

6.3 and prints its table in below format:

```
2*1=2
2*2=4
…….
2*10=20
```

```
n=$1
for((i=1;i<=10;i++))
do
echo "$n * $i = $(($i*$n))"
done
```

7.1 Write a menu- driven program for the following options

a. Accept a file name, starting and ending line numbers as arguments and display all the lines between the given line numbers from the file.
b. Check whether the file has write permission or not, if there is write permission Append two more lines to that file.

```
echo "enter your choice"
echo "1"
echo "2"
read ch
echo " enter your file "
read file1
if((ch==1))
then
echo " read start "
read start
echo " read end "
read end
end=$((end-1))
head -$end $file1 | tail -$((end-start))
else
if [ -w $file1 ]
then
 echo " it has write permission "
echo line1>>$file1
echo line2>>$file1
cat $file1
else
 echo " it has no permission "
fi

fi
```

7.2 Write a shell program to find largest among three numbers

```
echo “ EnterNum1 ”
read num1
echo “ EnterNum2 ”
read num2
echo “ EnterNum 3”
read num3
if [ $num1 -gt $num2 ] && [ $num1 -gt $num3 ]
then
echo " $num1 is largest "
elif [ $num2 -gt $num1 ] && [ $num2 -gt $num3 ]
then
echo " $num2 is largest "
else
echo " $num3 is largest "
fi
```

8.1 Write a shell script to save all usernames and their id number in your system to a file userid.

```
cut -d : -f 1,3 /etc/passwd>file1.txt
```

8. .2 Write a shell script to create a simple calculator

```
echo "Enter the 1st number"
read a
echo "Enter the 2nd number"
read b
echo "1. Addition"
echo "2. Substraction"
echo "3. Multiplication"
echo "4. Division"
read x
if [ $x -eq '1' ]
then
echo "SUM: $((a+b))"
elif [ $x -eq '2' ]
```

```
then
echo "SUBSTRACTION: $((a-b))"
elif [ $x -eq '3' ]
then
echo "Product: $((a*b))"
else [ $x -eq '4' ]
echo "Division: $((a/b))"
fi
```