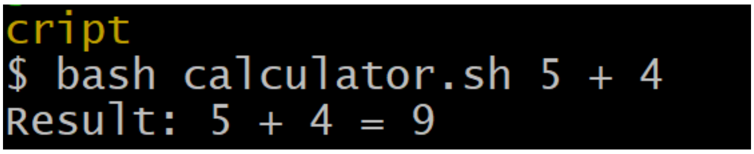
Shell Script

1. **For the Calculator use command line arguments:**

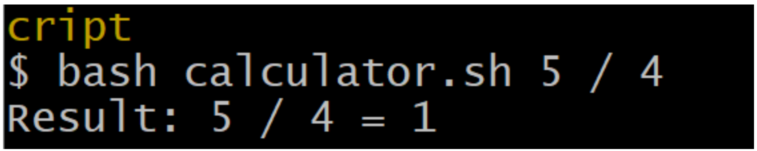
Code:

```
n1=$1
operator=$2
n2=$3
case $operator in
  +) result=$(($n1 + $n2)) ;;
  -) result=$(($n1 - $n2)) ;;
  \*) result=$(($n1 * $n2)) ;;
  /) if [ $n2 -eq 0 ];
      then
      echo "Division by zero is not allowed."
      exit 1
     fi
     result=$(($n1 / $n2)) ;;
  *) echo "Invalid operator: $operator"
     exit 1 ;;
esac
echo "Result: $n1 $operator $n2 = $result"
```

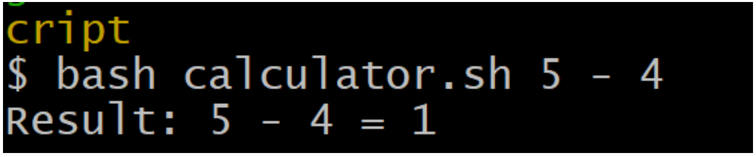
Output:



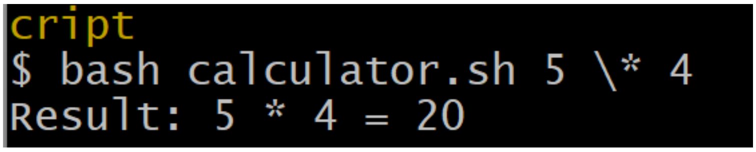
```
cript
$ bash calculator.sh 5 + 4
Result: 5 + 4 = 9
```



```
cript
$ bash calculator.sh 5 / 4
Result: 5 / 4 = 1
```



```
cript
$ bash calculator.sh 5 - 4
Result: 5 - 4 = 1
```



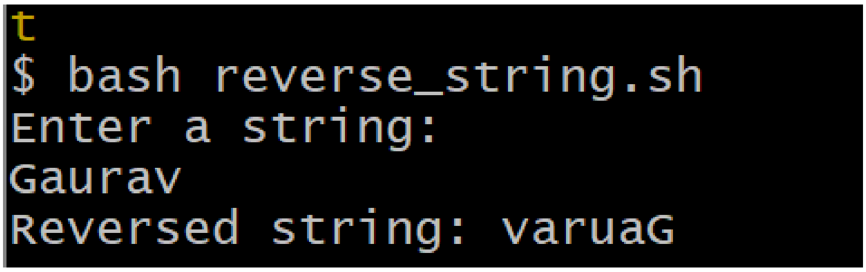
```
cript
$ bash calculator.sh 5 \* 4
Result: 5 * 4 = 20
```

2. **To reverse the given string**

Code:

```
echo "Enter a string:"

read string


reversed_string=""

length=${#string}

for (( i=$length-1; i>=0; i-- )); do

  reversed_string+=${string:$i:1}

done

echo "Reversed string: $reversed_string"
```

Output:



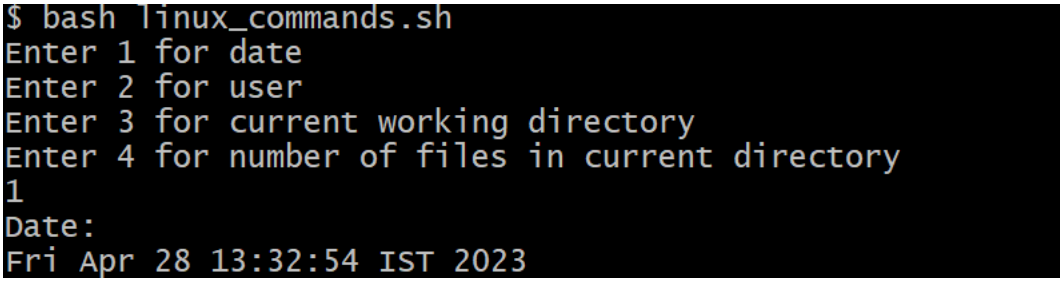
```
t
$ bash reverse_string.sh
Enter a string:
Gaurav
Reversed string: varuaG
```

3. **To execute Linux commands using case statements.**

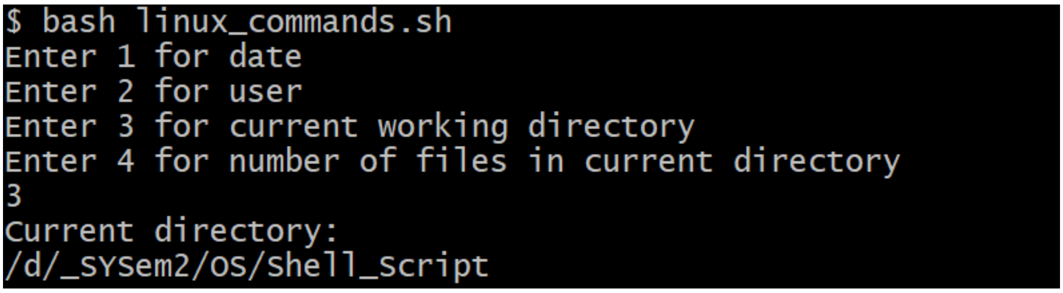
Code:

```
echo "Enter 1 for date"

echo "Enter 2 for user"

echo "Enter 3 for current working directory"

echo "Enter 4 for number of files in current directory"

read operation

case $operation in

   1) echo "Date: "

      date

      ;;

   2) echo "Username: "

      whoami

      ;;

   3) echo "Current directory: "

      pwd

      ;;

   4)  echo "Number of files: "

      ls  | wc -l

      ;;

   *)

      echo "Invalid operation"

      exit 1

      ;;

Esac
```

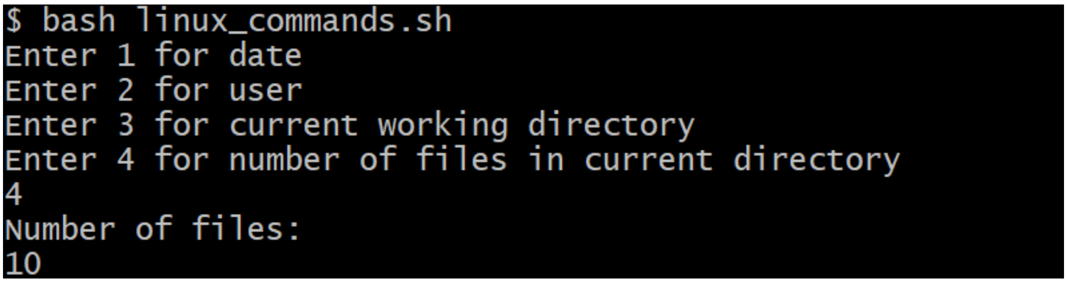
Output:



```
$ bash linux_commands.sh
Enter 1 for date
Enter 2 for user
Enter 3 for current working directory
Enter 4 for number of files in current directory
1
Date:
Fri Apr 28 13:32:54 IST 2023
```



```
$ bash linux_commands.sh
Enter 1 for date
Enter 2 for user
Enter 3 for current working directory
Enter 4 for number of files in current directory
3
Current directory:
/d/_SYSem2/OS/Shell_Script
```



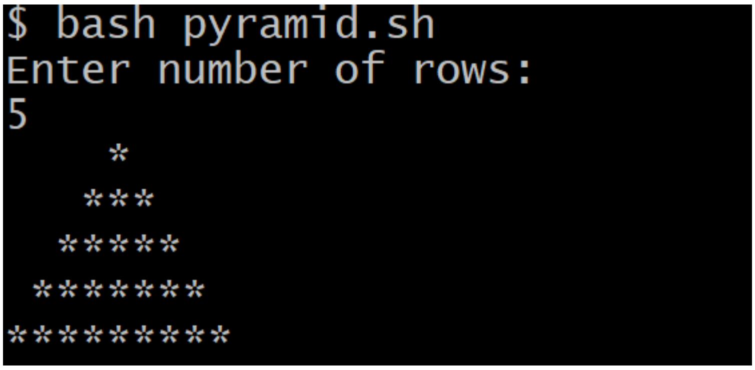
```
$ bash linux_commands.sh
Enter 1 for date
Enter 2 for user
Enter 3 for current working directory
Enter 4 for number of files in current directory
4
Number of files:
10
```

4. To print the pyramid of *

Code:

```
echo "Enter number of rows:"
read rows
for ((i=1; i<=rows; i++));
 do
 for ((j=1; j<=rows-i; j++));
  do
  echo -n " "
  done
  for ((k=1; k<=2*i-1; k++));
  do
  echo -n "*"
  done
 echo ""
done
```

Output:



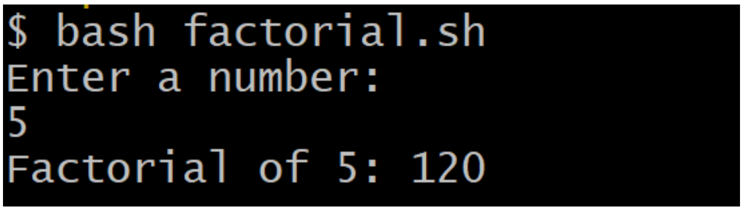
```
$ bash pyramid.sh
Enter number of rows:
5
    *
   ***
  *****
 *******
*********
```

5. To write a function for the factorial of a number

Code:

```
factorial() {
 local n=$1
 if [ $n -eq 0 ];
   then
   echo 1
 else
   echo $(( $n * $(factorial $((n-1))) ))
 fi
}

echo "Enter a number:"
read number

result=$(factorial $number)

echo "Factorial of $number: $result"
```

Output:



```
$ bash factorial.sh
Enter a number:
5
Factorial of 5: 120
```

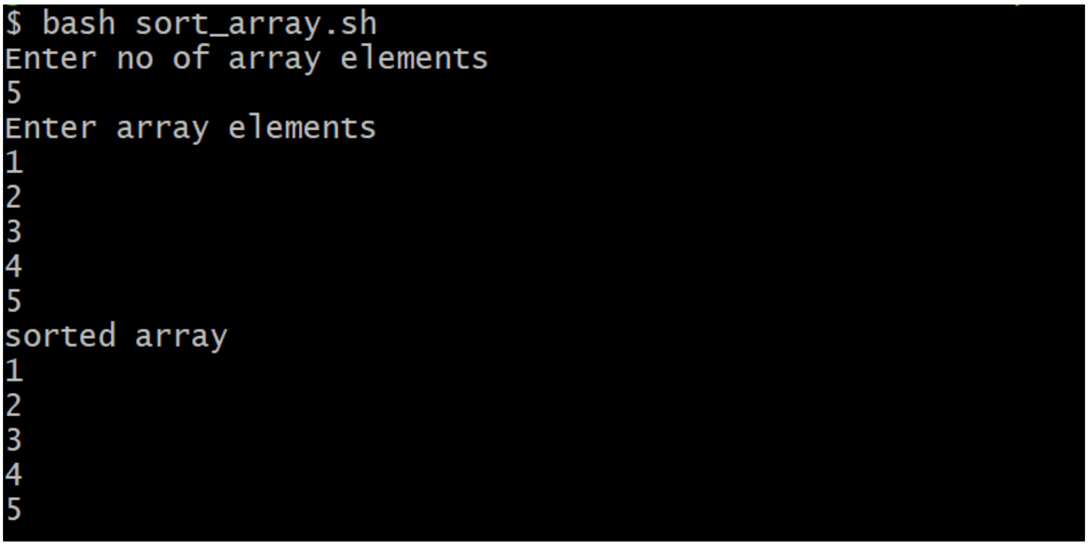
6. To sort the given elements using any sorting method.

Code:

```
echo "Enter no of array elements"
read n

echo "Enter array elements"

for((i=0;i<n;i++));
do
  read arr[$i]
done

for((i=0; i<n; i++));
do
  for((j=$i + 1; j<n; j++));
  do
    if [ ${arr[$i]} -gt ${arr[$j]} ]; then
    t=${arr[$j]}
    arr[$j]=${arr[$i]}
    arr[$i]=$t
    fi
  done
done
```

```
echo "sorted array"

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

do

   echo ${arr[$i]}

done
```

Output:



```
$ bash sort_array.sh
Enter no of array elements
5
Enter array elements
1
2
3
4
5
sorted array
1
2
3
4
5
```