

1. Write shell scripts for the following:

a. To take your name, programme name and enrolment number as input from user and print it on the screen.

```
echo "Enter your name : "  
read name  
  
echo "Programme : "  
read prg  
  
echo "Enrolment Number : "  
read eno  
  
printf "Name : $name \nProgramme : $prg\nEnrollement Number : $eno\n"
```

```
Enter your name :  
Amrita  
Programme :  
CSE  
Enrolment Number :  
23456  
Name : Amrita  
Programme : CSE  
Enrollement Number : 23456
```

b. To find the sum, the average and the product of four integers.

```
echo "Enter four numbers "  
read a b c d  
  
sum=$((a+b+c+d))  
avg=$((sum/4))  
product=$((a*b*c*d))  
printf "Sum : $sum\nAverage : $avg\nProduct : $product\n"
```

```
Enter four numbers  
2 3 4 5  
Sum : 14  
Average : 3  
Product : 120
```

c. Write a program to check whether a number is even or odd.

```
echo "enter a number : "  
read n  
if [  $$(n\%2)$  -eq 0 ]  
then  
    echo "$n is Even"  
else  
    echo "$n is Odd"  
fi
```

```
enter a number :  
66  
66 is Even
```

d. To exchange the values of two variables.

```
echo "Enter two numbers : "  
read a b  
printf "Before Swaapping : $a,$b "  
temp=$a  
a=$b  
b=$temp  
printf "After Swapping : $a , $b "
```

```
Enter two numbers :  
2 3  
Before Swaapping : 2,3  
After Swapping : 3 , 2  
Enter file name:  
_
```

e. To find the lines containing a number in a file.

```
echo "Enter file name:"  
read file  
grep '[0-9]' "$file"
```

```
Enter file name:
file
one 1
two 2
three 3
```

f. To concatenate two strings and find the length of the resultant string.

```
echo "Enter first string : "
read str1
echo "Enter second string : "
read str2
str=$str1$str2
echo "Concatenated String : $str"
echo "Length of string : ${#str}"
```

```
Enter first string :
Hello
Enter second string :
All !!!
Concatenated String : HelloAll !!!
Length of string : 12
```

g. To concatenate the contents of two files.

```
echo "Enter file names : "
read a b
cat "$a" "$b" > merged_file
echo "content of $a and $b"
cat merged_file
```

```
Enter file names :
file1 file2
content of file1 and file2
This is from file1
This is from file2
```

h. Write a shell script that would wait 5 seconds and then display the time.

```
echo "Waiting for 5 seconds"
sleep 5
echo "Current time : $(date)"
```

Waiting for 5 seconds

Current time : Mon Mar 3 08:05:42 PM UTC 2025

2. The length and breadth of a rectangle and radius of a circle are provided as user input. Write a shell script that will calculate the area and perimeter of the rectangle and the area and circumference of the circle.

Hint:- Area of Rectangle =  $L \times B$  Perimeter of Rectangle =  $2(L+B)$  Area of Circle =  $\pi \cdot r^2$  Circumference of circle =  $2 \cdot \pi \cdot r$

```
echo "Length ,Breath of Rectangle :"  
read l b  
echo "Radius of Circle :"  
read r  
  
area_cir=$(echo "3.14*$r*$r" |bc)  
circumfence_cir=$(echo "2*3.14*$r" |bc)  
  
area_rect=$(echo "$l*$b" |bc)  
perimeter_rect=$(echo "2*($l+$b)" |bc)  
  
echo -e "Rectangle\nArea=$area_rect Perimeter=$perimeter_rect\n"  
echo -e "Circle\nArea=$area_cir, circumference=$circumfence_cir"
```

```
Length ,Breath of Rectangle :  
2.5 3.2  
Radius of Circle :  
3  
Rectangle  
Area=8.0 Perimeter=11.4  
  
Circle  
Area=28.26, circumference=
```

3. Write a menu driven shell program to read two numbers and print the results of all the arithmetic operations. ( + , - , \* , / , % , ++ , -- )

```
printf "$num1 + $num2 : $((num1 + num2))\n"
printf "$num1 - $num2 : $((num1 - num2))\n"
printf "$num1 * $num2 : $((num1 * num2))\n"
printf "$num1 / $num2 : $((num1 / num2))\n"
printf "$num1 %% $num2 : $((num1 % num2))\n"
printf "num1++ : $num1\n"
((num1++))
printf "After num1++ : $num1\n"
printf "num2++ : $num2\n"
((num2++))
printf "After num2++ : $num2\n"
printf "num1-- : $num1\n"
((num1--))
printf "After num1-- : $num1\n"
printf "num2-- : $num2\n"
((num2--))
printf "After num2-- : $num2\n"
```

Enter 2 numbers:

```
2 3
2 + 3 : 5
2 - 3 : -1
2 * 3 : 6
2 / 3 : 0
2 % 3 : 2
num1++ : 2
After num1++ : 3
num2++ : 3
After num2++ : 4
num1-- : 3
After num1-- : 2
num2-- : 4
After num2-- : 3
```

4. Write two separate shell scripts to find the factorial of a number using while statement and for statement.

```
#Factorial using while loop
echo "Enter a number:"
read num
fact=1
while [ $num -gt 0 ]
do
    fact=$((fact * num))
    num=$((num - 1))
done
echo "Factorial using while loop : $fact"

#Factorial using for loop
fact=1
for ((i=1; i<=num; i++))
do
    fact=$((fact * i))
done
echo "Factorial using for loop : $fact"
```

```
Enter a number:
5
Factorial using while loop : 120
Enter a number:
5
Factorial using for loop : 120
```

5. Given a file of numbers (one number per line), write a shell script that will find the lowest and highest number.

```

echo "Enter file name : "
read file
if [ ! -f "$file" ]; then
    echo "File not found!"
    exit 1
fi
min=$(head -n 1 $file)
max=$min
while read num
do
    if [ $num -lt $min ]; then
        min=$num
    fi
    if [ $num -gt $max ]; then
        max=$num
    fi
done < "$file"
echo "Lowest number: $min"
echo "Highest number: $max"

```

```

Enter file name :
nums
Lowest number: 1
Highest number: 84

```

6. Write a shell program to read n numbers into an array and display the average of them

```

echo "Enter the number of elements:"
read n
sum=0
for ((i=0; i<n; i++))
do
    echo "Enter number $((i+1)):"
    read num
    arr[$i]=$num
    sum=$((sum + num))
done
avg=$((sum / n))
echo "Average: $avg"

```



```

Enter the number of elements:
5
Enter number 1:
1
Enter number 2:
2
Enter number 3:
3
Enter number 4:
4
Enter number 5:
5
Average: 3

```

7a.

```

echo "Enter the number of rows:"
read rows

# Loop to print the left-down pyramid
for ((i=rows; i>=1; i--))
do
    for ((j=1; j<=i; j++))
    do
        echo -n "*"
    done
    echo
done

```

```

Enter the number of rows:
5
*****
****
***
**
*

```

b.

```

echo "Enter the 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

```



```
root@eggshead: ~ # ./lab7.sh
Enter the number of rows:
5
 *
 ***
 *****
 *******
 *******
 *******
```

8. Write a shell program to read two matrices, add them and print the output matrix.

```
echo "Enter number of rows and columns:"
read rows cols

echo "Enter elements of first matrix:"
for ((i=0; i<rows; i++))
do
    for ((j=0; j<cols; j++))
    do
        echo -n "Enter element at position ($i, $j): "
        read matrix1[$i,$j]
    done
done

echo "Enter elements of second matrix:"
for ((i=0; i<rows; i++))
do
    for ((j=0; j<cols; j++))
    do
        echo -n "Enter element at position ($i, $j): "
        read matrix2[$i,$j]
    done
done

echo "The sum of the matrices is:"
for ((i=0; i<rows; i++))
do
    for ((j=0; j<cols; j++))
    do
        sum=$((matrix1[$i,$j] + matrix2[$i,$j]))
        result[$i,$j]=$sum
        echo -n "${result[$i,$j]} "
    done
    echo
done
```

```

Enter number of rows and columns:
2 2
Enter elements of first matrix:
Enter element at position (0, 0): 1
Enter element at position (0, 1): 2
Enter element at position (1, 0): 3
Enter element at position (1, 1): 4
Enter elements of second matrix:
Enter element at position (0, 0): 5
Enter element at position (0, 1): 6
Enter element at position (1, 0): 7
Enter element at position (1, 1): 8
The sum of the matrices is:
10 12
10 12

```

9. Write a program to read a matrix and print the transpose of it

```

echo "Enter number of rows and columns:"
read rows cols
echo "Enter elements of the matrix:"
for ((i=0; i<rows; i++))
do
    for ((j=0; j<cols; j++))
    do
        echo -n "Enter element at position ($i, $j): "
        read matrix[$i,$j]
    done
done
echo "The original matrix is:"
for ((i=0; i<rows; i++))
do
    for ((j=0; j<cols; j++))
    do
        echo -n "${matrix[$i,$j]} "
    done
    echo
done
echo "The transpose of the matrix is:"
for ((i=0; i<cols; i++))
do
    for ((j=0; j<rows; j++))
    do
        echo -n "${matrix[$j,$i]} "
    done
    echo
done

```

```

Enter number of rows and columns:
2 3
Enter elements of the matrix:
Enter element at position (0, 0): 1
Enter element at position (0, 1): 2
Enter element at position (0, 2): 3
Enter element at position (1, 0): 4
Enter element at position (1, 1): 5
Enter element at position (1, 2): 6

```

The original matrix is:

1 2 3

4 5 6

The transpose of the matrix is:

1 4

2 5

3 6