LAB ASSIGNMENT - 3

Aim: To create shell scripts for the following questions

To perform: To code and solve the following

Q1. To find Largest of Three Numbers

```
#!/bin/bash
echo “Enter first number:”
read a
echo “Enter second number:”
read b
echo “Enter third number:”
read c
if [ $a -ge $b ] && [ $a -ge $c ]
then
   echo "Largest number is: $a"
elif [ $b -ge $a ] && [ $b -ge $c ]
 then
   echo "Largest number is: $b"
else
   echo "Largest number is: $c"
fi
```

Q2. To find a year is leap year or not.

```
#!/bin/bash
echo “Enter year:”
read year
if (((year % 4 == 0 && year % 100!= 0) || (year % 400 == 0) ))
then
    echo ”Leap year”
```

```
else
  echo “Not a leap year”
fi
```

Q3. To input angles of a triangle and find out whether it is valid triangle or not

```
#!/bin/bash
echo "Enter three angles of a triangle: "
read x y z
sum=$((x + y + z))
if [ $sum -eq 180 ] && [ $x -gt 0 ] && [ $y -gt 0 ] && [ $z -gt 0 ]
 then
   echo "Valid Triangle"
else
   echo "Invalid Triangle"
fi
```

Q4. To check whether a character is alphabet, digit or special character.

```
#!/bin/bash
echo "Enter a character: "
read char
if [[ $char =~ [a-zA-Z] ]]
then
   echo "Alphabet"
elif [[ $char =~ [0-9] ]]
then
   echo "Digit"
else
   echo "Special Character"
fi
```

Q5. To calculate profit or loss

```
#!/bin/bash
echo "Enter cost price and selling price: "
read cp sp
if [ $sp -gt $cp ]
then
   echo "Profit of rupees $((sp - cp))"
elif [ $sp -lt $cp ]
then
   echo "Loss of rupees $((cp - sp))"
else
   echo "No Profit No Loss"
fi
```

Q6. To print all even and odd number from 1 to 10

```
#!/bin/bash
echo "Even numbers:"
for i in {2..10..2}
do
   echo $i
done
echo "Odd numbers:"
for(( i=1; i<10; i+=2))
do
   echo $i
done
```

Q7. To print table of a given number

```
#!/bin/bash
echo "Enter a number:"
read n
for ((i=1;i<=10; i++))
do
  echo “$n x $i = $((n*i ))”
done
```

Q8. To find factorial of a given integer

```
#!/bin/bash
echo "Enter a number:"
read n
fact=1
for ((i=1;i<=10; i++))
do
  ((fact=fact*n))
done
echo $fact
```

Q9. To print sum of all even numbers from 1 to 10.

```
#!/bin/bash
sum=0
for ((i=1;i<=10; i++))
do
   if ((i%2==0))
   then
       ((sum+=i))
   fi
done
echo $sum
```

Q10. To print sum of digit of any number.

```
#!/bin/bash
echo "Enter a number: "
read num
sum=0
while [ $num -gt 0 ];
do
   sum=$((sum + num % 10))
   num=$((num / 10))
done
echo$sum
```

Q11. To make a basic calculator which performs addition, subtraction, Multiplication, Division

```
echo "Enter two numbers: "
read a b
echo "Enter operation (+, -, *, /): "
read op
echo "Result: $(echo "$a $op $b" | bc)"
```

Q12. Print Days of a Week

```
echo "Monday Tuesday Wednesday Thursday Friday Saturday Sunday"
```

Q)13. Print First 4 Months Having 31 Days

```
echo "January March May July"
```

Q14. Using functions,

a. To find given number is Amstrong number or not

```
is_armstrong() {
   num=$1
   sum=0
   temp=$num
   while [ $temp -gt 0 ]
   do
      digit=$((temp % 10))
```

```
    sum=$((sum + digit ** 3))
    temp=$((temp / 10))
  done
  if [ $sum -eq $num ]
   then
    echo "$num is an Armstrong Number"
  else
    echo "$num is not an Armstrong Number"
  fi
}
```

b. To find whether a number is palindrome or not

```
is_palindrome() {
  num=$1
  rev=0
  temp=$num
  while [ $temp -gt 0 ]
  do
    digit=$((temp % 10))
    rev=$((rev * 10 + digit))
    temp=$((temp / 10))
  done
  if [ $rev -eq $num ]
  then
    echo "$num is a Palindrome"
  else
    echo "$num is not a Palindrome"
  fi
}
```

c. To print Fibonacci series upto n terms

```
fibonacci() {
  n=$1
```

```
    first=0
    second=1
    echo -n "$a $b "
    for ((i=2; i<n; i++))
    do
        c=$((a + b))
        echo -n "$c "
        a=$b
        b=$c
    done
    echo
}
```

d. To find given number is prime or composite

```
isPrime(){
    num=$1
    for ((i=2; i*i<=num; i++))
    do
        if [ $((num % i)) -eq 0 ]
          then
           echo "$num is a Composite Number"
           return
        fi
    done
    echo "$num is a Prime Number"
}
```

e. To convert a given decimal number to binary equivalent

```
decimal_to_binary() {
    num=$1
    echo "obase=2; $num" | bc
}
```