Lab Assignment 3

1. To find Largest of Three Numbers

Ans. #!/bin/bash

```
# Script to find the largest of three numbers

echo "Enter first number:"
read a
echo "Enter second number:"
read b
echo "Enter third number:"
read c

if [ $a -ge $b ] && [ $a -ge $c ]; then
    echo "$a is the largest"
elif [ $b -ge $a ] && [ $b -ge $c ]; then
    echo "$b is the largest"
```

```
else
    echo "$c is the largest"
fi
```

2. To find a year is leap year or not

Ans.

```
#!/bin/bash

# Script to check if a year is a leap year

echo "Enter a year:"
read year

if (( year % 400 == 0 )); then
    echo "$year is a leap year"
elif (( year % 100 == 0 )); then
    echo "$year is not a leap year"
elif (( year % 4 == 0 )); then
    echo "$year is a leap year"
else
```

```
    echo "$year is not a leap year"
fi
```

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

Ans.

```
#!/bin/bash

# Input angles of a triangle and check if it's valid

echo "Enter first angle:"
read a
echo "Enter second angle:"
read b
echo "Enter third angle:"
read c

sum=$((a + b + c))
```

```
if [ $sum -eq 180 ] && [ $a -gt 0 ] && [ $b -gt 0 ] && [ $c -gt 0 ]; then
    echo "It is a valid triangle"
else
    echo "It is NOT a valid triangle"
fi
```

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

Ans.

```
#!/bin/bash

# Check if input character is alphabet, digit, or special character

echo "Enter a character:"
read ch

if [[ $ch =~ [A-Za-z] ]]; then
    echo "It is an alphabet"
```

```
elif [[ $ch =~ [0-9] ]]; then
    echo "It is a digit"
else
    echo "It is a special character"
fi
```

5. To calculate profit or loss

Ans.

```
#!/bin/bash

# Calculate profit or loss

echo "Enter Cost Price:"
read cp
echo "Enter Selling Price:"
read sp

if (( sp > cp )); then
    profit=$((sp - cp))
    echo "Profit: ₹$profit"
```

```
elif (( sp < cp )); then
    loss=$((cp - sp))
    echo "Loss: ₹$loss"
else
    echo "No profit, no loss"
fi
```

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

Ans.

```
#!/bin/bash

echo "Even numbers from 1 to 10:"
for ((i=1; i<=10; i++)); do
    if (( i % 2 == 0 )); then
        echo -n "$i "
    fi
done

echo -e "\nOdd numbers from 1 to 10:"
for ((i=1; i<=10; i++)); do
```

```
    if (( i % 2 != 0 )); then
        echo -n "$i "
    fi
done
```

7. To print table of a given number

Ans.

```
#!/bin/bash

echo "Enter a number:"
read num

echo "Table of $num:"
for ((i=1; i<=10; i++)); do
    echo "$num x $i = $((num * i))"
done
```

8. To find factorial of a given integer

Ans.

```
#!/bin/bash

echo "Enter a number:"
```

```
read n
fact=1

for ((i=1; i<=n; i++)); do
    fact=$((fact * i))
done

echo "Factorial of $n is $fact"
```

9. To print sum of all even numbers from 1 to

Ans.

```
#!/bin/bash

sum=0
for ((i=1; i<=10; i++)); do
    if (( i % 2 == 0 )); then
        sum=$((sum + i))
    fi
done
```

```
echo "Sum of even numbers from 1 to 10 is: $sum"
```

10. 10. To print sum of digit of any number.

Ans.

```
#!/bin/bash

echo "Enter a number:"
read num
sum=0

while [ $num -gt 0 ]; do
    digit=$((num % 10))
    sum=$((sum + digit))
    num=$((num / 10))
done

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

11. To make a basic calculator which performs addition, subtraction, Multiplication, division

Ans. #!/bin/bash

```
echo "Enter first number:"
read a
echo "Enter second number:"
read b

echo "Choose operation: + - * /"
read op

case $op in
   +) result=$((a + b)) ;;
   -) result=$((a - b)) ;;
   \*) result=$((a * b)) ;;
   /)
      if [ $b -ne 0 ]; then
         result=$((a / b))
      else
```

```
            echo "Division by zero not allowed"
            exit
        fi
        ;;
    *) echo "Invalid operator"; exit ;;
esac

echo "Result: $result"
```

12. To print days of a week.

Ans.

```
#!/bin/bash

echo "Days of the week:"
days=("Sunday" "Monday" "Tuesday" "Wednesday" "Thursday" "Friday" "Saturday")

for day in "${days[@]}"; do
    echo "$day"
done
```

13. To print starting 4 months having 31 days.

Ans. #!/bin/bash

```
echo "Starting 4 months having 31 days:"
months=("January" "March" "May" "July")

for month in "${months[@]}"; do
    echo "$month"
done
```

14. Using functions

a. To find given number is Amstrong number or not

Ans. #!/bin/bash

```
is_armstrong() {
    num=$1
    sum=0
    temp=$num
```

```
    while [ $temp -gt 0 ]; do
        digit=$((temp % 10))
        sum=$((sum + digit * digit * digit))
        temp=$((temp / 10))
    done

    if [ $sum -eq $num ]; then
        echo "$num is an Armstrong number"
    else
        echo "$num is not an Armstrong number"
    fi
}

echo "Enter a number:"
read n
is_armstrong $n
```

b. To find whether a number is palindrome or not

Ans. #!/bin/bash

```
is_palindrome() {
    num=$1
    reverse=0
    temp=$num

    while [ $temp -gt 0 ]; do
        digit=$((temp % 10))
        reverse=$((reverse * 10 + digit))
        temp=$((temp / 10))
    done

    if [ $reverse -eq $num ]; then
        echo "$num is a palindrome"
    else
```

```
        echo "$num is not a palindrome"
    fi
}

echo "Enter a number:"
read n
is_palindrome $n
```

c. To print Fibonacci series upto n terms

Ans.

```
#!/bin/bash

fibonacci() {
    n=$1
    a=0
    b=1

    echo "Fibonacci series up to $n terms:"
    for ((i=0; i<n; i++)); do
        echo -n "$a "
```

```
        fn=$((a + b))
        a=$b
        b=$fn
    done
    echo
}

echo "Enter number of terms:"
read n
fibonacci $n
```

d. To find given number is prime or composite e. To convert a given decimal number to binary equivalent

Ans. #!/bin/bash

```
is_prime() {
    num=$1
```

```
    if [ $num -le 1 ]; then
        echo "$num is not a prime number"
        return
    fi

    for ((i=2; i*i<=num; i++)); do
        if ((num % i == 0)); then
            echo "$num is a composite number"
            return
        fi
    done

    echo "$num is a prime number"
}

echo "Enter a number:"
read n
is_prime $n
```