OPERATING SYSTEMS

LAB ASSIGNMENT 3

Give shell scripts for following:

1. To find Largest of Three Numbers.

```
#!/bin/bash

echo "Enter first number:"
read num1

echo "Enter second number:"
read num2

echo "Enter third number:"
read num3

if [ $num1 -ge $num2 ] && [ $num1 -ge $num3 ]; then
   echo "The largest number is: $num1"
elif [ $num2 -ge $num1 ] && [ $num2 -ge $num3 ]; then
   echo "The largest number is: $num2"
else
   echo "The largest number is: $num3"
fi
```

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

```
#!/bin/bash

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.

```
#!/bin/bash
```

```
echo "Enter first angle:"
read angle1

echo "Enter second angle:"
read angle2

echo "Enter third angle:"
read angle3

sum=$((angle1 + angle2 + angle3))

if (( sum == 180 && angle1 > 0 && angle2 > 0 && angle3 > 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.

```
#!/bin/bash

echo "Enter a character:"
read char

if [[ $char =~ [A-Za-z] ]]; then
   echo "It is an alphabet"
elif [[ $char =~ [0-9] ]]; then
   echo "It is a digit"
else
   echo "It is a special character"
fi
```

5. To calculate profit or loss.

```
#!/bin/bash

echo "Enter cost price:"
read cp

echo "Enter selling price:"
read sp

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

```
elif (( cp > sp )); 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.

```
#!/bin/bash

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

echo "Odd numbers:"
for (( i=1; i<=10; i++ ))
do
   if (( i % 2 != 0 )); then
      echo "$i"
   fi
done
```

7. To print table of a given number.

```
#!/bin/bash

echo "Enter a number:"
read num

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

8. To find factorial of a given integer.

```
#!/bin/bash

echo "Enter a number:"
read num
```

```
fact=1

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

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

9. 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=$((sum + i))
   fi
done

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

10. To print sum of digit of any number.

```
#!/bin/bash

echo "Enter a number:"
read num

sum=0

while (( num > 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.

```
#!/bin/bash

echo "Enter first number:"
read a

echo "Enter second number:"
read b

echo "Choose operation (+, -, *, /):"
read op

case $op in
   +) echo "Result = $((a + b))" ;;
   -) echo "Result = $((a - b))" ;;
   \*) echo "Result = $((a * b))" ;;
   /)
      if (( b == 0 )); then
         echo "Division by zero not allowed"
      else
         echo "Result = $((a / b))"
      fi
      ;;
   *) echo "Invalid operator" ;;
esac
```

12. To print days of a week.

```
#!/bin/bash

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.

```
#!/bin/bash

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

```
#!/bin/bash

amstrong() {
   echo "Enter a number:"
   read num
   n=$num
   sum=0
   while (( n > 0 ))
   do
      digit=$((n % 10))
      sum=$((sum + digit * digit * digit))
      n=$((n / 10))
   done
   if (( sum == num )); then
      echo "$num is an Armstrong number"
   else
      echo "$num is not an Armstrong number"
   fi
}

amstrong
```

b. To find whether a number is palindrome or not

```
#!/bin/bash

palindrome() {
   echo "Enter a number:"
   read num
   original=$num
   rev=0
   while (( num > 0 ))
   do
      digit=$((num % 10))
      rev=$((rev * 10 + digit))
      num=$((num / 10))
   done
   if (( rev == original )); then
      echo "$original is a palindrome"
   else
      echo "$original is not a palindrome"
```

```
    fi
}

palindrome
```

c. To print Fibonacci series upto n terms

```
#!/bin/bash

fibonacci() {
   echo "Enter number of terms:"
   read n
   a=0
   b=1
   echo "Fibonacci series:"
   for (( i=0; i<n; i++ ))
   do
      echo -n "$a "
      fn=$((a + b))
      a=$b
      b=$fn
   done
   echo
}

fibonacci
```

d. To find given number is prime or composite

```
#!/bin/bash

prime() {
   echo "Enter a number:"
   read num
   if (( num <= 1 )); then
      echo "$num is neither prime nor composite"
      return
   fi
   for (( i=2; i*i<=num; i++ ))
   do
      if (( num % i == 0 )); then
         echo "$num is a composite number"
         return
      fi
   done
   echo "$num is a prime number"
```

```
}

prime
```

e. To convert a given decimal number to binary equivalent

```
#!/bin/bash

decimal_to_binary() {
   echo "Enter a decimal number:"
   read num
   binary=""
   n=$num
   if (( n == 0 )); then
      binary="0"
   else
      while (( n > 0 ))
      do
         rem=$((n % 2))
         binary="$rem$binary"
         n=$((n / 2))
      done
   fi
   echo "Binary of $num is $binary"
}

decimal_to_binary
```