4ITRC2 Operating System Lab

Lab Assignment 3

Aim: To create shell scripts for the following questions

To perform: To code and solve the following

To Submit: Give shell scripts for following:

1. **To find Largest of Three Numbers**

```
echo "Enter three numbers:"
read a b 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
```

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

```
echo "Enter a year:"
read year
if [ $((year % 4)) -eq 0 ] && ([ $((year % 100)) -ne 0 ] || [ $((year % 400)) -
eq 0 ]); then
   echo "$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**

```
echo "Enter three angles of a triangle:"
read a b c
sum=$((a + b + c))
if [ $sum -eq 180 ]; then
   echo "Valid triangle"
else
   echo "Invalid triangle"
fi
```

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

```
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
```

5. **To calculate profit or loss**

```
echo "Enter cost price:"
read cost
echo "Enter selling price:"
read selling
if [ $selling -gt $cost ]; then
   echo "Profit: $((selling - cost))"
elif [ $cost -gt $selling ]; then
   echo "Loss: $((cost - selling))"
else
   echo "No Profit, No Loss"
fi
```

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

```
echo "Even numbers:"
for i in {1..10}; do
   if [ $((i % 2)) -eq 0 ]; then
      echo $i
   fi
done

echo "Odd numbers:"
for i in {1..10}; do
   if [ $((i % 2)) -ne 0 ]; then
      echo $i
   fi
```

7. **To print table of a given number**

```
echo "Enter a number:"
read n
for i in {1..10}; do
   echo "$n x $i = $((n * i))"
```

8. **To find factorial of a given integer**

```
echo "Enter a number:"
read n
factorial=1
for ((i = 1; i <= n; i++)); do
   factorial=$((factorial * i))
done
echo "Factorial: $factorial"
```

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

```
sum=0
for i in {1..10}; do
   if [ $((i % 2)) -eq 0 ]; then
      sum=$((sum + i))
   fi
done
echo "Sum of even numbers: $sum"
```

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

```
echo "Enter a number:"
read n
sum=0
while [ $n -gt 0 ]; do
   digit=$((n % 10))
   sum=$((sum + digit))
   n=$((n / 10))
done
echo "Sum of digits: $sum"
```

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

```
echo "Enter two numbers:"
read a b
echo "Choose operation: 1-Add 2-Subtract 3-Multiply 4-Divide"
read choice
case $choice in
1) echo "Result: $((a + b))" ;;
2) echo "Result: $((a - b))" ;;
3) echo "Result: $((a * b))" ;;
4) echo "Result: $((a / b))" ;;
*) echo "Invalid choice" ;;
Esac
```

12. **To print days of a week.**

```
for day in Monday Tuesday Wednesday Thursday Friday Saturday Sunday;
do
   echo $day
done
```

13. **To print starting 4 months having 31 days.**

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

14. **Using functions,**
    a. **To find given number is Amstrong number or not**

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

```
    sum=0
    original=$n
    while [ $n -gt 0 ]; do
        digit=$((n % 10))
        sum=$((sum + digit ** 3))
        n=$((n / 10))
    done
    if [ $sum -eq $original ]; then
        echo "Armstrong Number"
    else
        echo "Not an Armstrong Number"
    fi
}
```

b. To find whether a number is palindrome or not

```
is_palindrome() {
    number=$1
    reversed=0
    original=$number
    while [ $number -gt 0 ]; do
        digit=$((number % 10))
        reversed=$((reversed * 10 + digit))
        number=$((number / 10))
    done
    if [ $reversed -eq $original ]; then
        echo "Palindrome Number"
    else
        echo "Not a Palindrome Number"
    fi
}
echo "Enter a number:"
read num
is_palindrome $num
```

c. To print Fibonacci series upto n terms

```
fibonacci() {
terms=$1
a=0
b=1
echo "Fibonacci Series:"
echo $a
echo $b
for ((i=3; i<=terms; i++)); do
    c=$((a + b))
    echo $c
    a=$b
    b=$c
```

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

d. To find given number is prime or composite

```
is_prime() {
   number=$1
   if [ $number -le 1 ]; then
      echo "Neither Prime nor Composite"
      return
   fi
   for ((i=2; i*i<=number; i++)); do
      if [ $((number % i)) -eq 0 ]; then
         echo "Composite Number"
         return
      fi
   done
   echo "Prime Number"
}
echo "Enter a number:"
read num
is_prime $num
```

e. To convert a given decimal number to binary equivalent

```
decimal_to_binary() {
   number=$1
   binary=""
   while [ $number -gt 0 ]; do
      remainder=$((number % 2))
      binary="$remainder$binary"
      number=$((number / 2))
   done
   echo "Binary Equivalent: $binary"
}
echo "Enter a decimal number:"
read num
decimal_to_binary $num
```