**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:

# Largest of Three Numbers

echo "Enter three numbers:" read a b 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

# Leap Year

echo "Enter a year:" read year

if (( (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0) )); then echo "$year is a Leap Year"

else

echo "$year is Not a Leap Year"

fi

# Valid Triangle Check

echo "Enter three angles of triangle:" read a b c

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

if [ $sum -eq 180 ] && [ $a -gt 0 ] && [ $b -gt 0 ] && [ $c -gt 0 ]; then

echo "Valid Triangle" else

echo "Invalid Triangle"

fi

# Character Type Check

echo "Enter a character:" read ch

case $ch in

[a-zA-Z]) echo "Alphabet";;

[0-9]) echo "Digit";;

\*) echo "Special Character";; esac

# Profit or Loss

echo "Enter Cost Price:" read cp

echo "Enter Selling Price:" read sp

if [ $sp -gt $cp ]; then profit=$((sp - cp)) echo "Profit: $profit"

elif [ $cp -gt $sp ]; then loss=$((cp - sp)) echo "Loss: $loss"

else

echo "No Profit No Loss"

fi

# Even and Odd 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 done

# Multiplication Table

echo "Enter a number:" read num

for i in {1..10} do

echo "$num x $i = $((num \* i))" done

# Factorial

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"

# Sum of Even Numbers (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 from 1 to 10 is $sum"

# Sum of Digits

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: $sum"

# Basic Calculator

echo "Enter two numbers:" read a b

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

case $op in

+) echo "Result: $((a + b))";;

-) echo "Result: $((a - b))";;

\\*) echo "Result: $((a \* b))";;

/) if [ $b -ne 0 ]; then

echo "Result: $((a / b))" else

echo "Cannot divide by zero" fi;;

\*) echo "Invalid operation";; esac

# Days of the Week

days=("Sunday" "Monday" "Tuesday" "Wednesday" "Thursday" "Friday" "Saturday")

for day in "${days[@]}" do

echo $day done

# First 4 Months with 31 Days

months=("January" "March" "May" "July") for month in "${months[@]}"

do

echo $month done

# Using Functions

is\_amstrong() { n=$1

sum=0 temp=$n

while [ $temp -gt 0 ]; do digit=$((temp % 10))

sum=$((sum + digit \* digit \* digit)) temp=$((temp / 10))

done

if [ $sum -eq $n ]; then echo "Amstrong Number"

else

echo "Not Amstrong"

fi

}

is\_palindrome() { n=$1

rev=0 temp=$n

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

done

if [ $rev -eq $n ]; then echo "Palindrome"

else

echo "Not Palindrome"

fi

}

fibonacci() { n=$1

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

}

is\_prime() { n=$1

if [ $n -le 1 ]; then echo "Not Prime" return

fi

for (( i=2; i\*i<=n; i++ )); do

if [ $((n % i)) -eq 0 ]; then echo "Composite"

return

fi done

echo "Prime"

}

dec\_to\_bin() { n=$1 bin=""

while [ $n -gt 0 ]; do bin=$((n % 2))$bin n=$((n / 2))

done

echo "Binary: $bin"

}