# 

# **Operating System Lab Assignment III**

### **4ITRC2**

*Purvanshi patidar*

*BE II Year*

*Information technology – ‘A’*

# Aim

To create shell scripts for the following questions

# To perform

To code and solve the following

# To Submit

Give shell scripts for following:

### To find Largest of Three Numbers

#!/bin/bash

echo "Enter three numbers:"

read num1 num2 num3

if [ $num1 -gt $num2 ] && [ $num1 -gt $num3 ]; then

echo "$num1 is the largest"

elif [ $num2 -gt $num1 ] && [ $num2 -gt $num3 ]; then

echo "$num2 is the largest"

else

echo "$num3 is the largest"

fi

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

#!/bin/bash

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

### 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 angle1 angle2 angle3

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

if [ $sum -eq 180 ] && [ $angle1 -gt 0 ] && [ $angle2 -gt 0 ] && [ $angle3 -gt 0 ]; then

echo "Valid triangle"

else

echo "Invalid triangle"

fi

### 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

### To calculate profit or loss

#!/bin/bash

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

### To print all even and odd number from 1 to 10

#!/bin/bash

echo "Even numbers from 1 to 10:"

for i in {1..10}; do

if [ $((i % 2)) -eq 0 ]; then

echo -n "$i "

fi

done

echo -e "\nOdd numbers from 1 to 10:"

for i in {1..10}; do

if [ $((i % 2)) -ne 0 ]; then

echo -n "$i "

fi

done

echo

### To print table of a given number

#!/bin/bash

echo "Enter a number:"

read num

for i in {1..10}; do

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

done

### 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"

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

#!/bin/bash

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: $sum"

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

#!/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: $sum"

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

#!/bin/bash

echo "Enter two numbers:"

read num1 num2

echo "Select operation:"

echo "1. Addition"

echo "2. Subtraction"

echo "3. Multiplication"

echo "4. Division"

read choice

case $choice in

1) echo "$num1 + $num2 = $((num1 + num2))" ;;

2) echo "$num1 - $num2 = $((num1 - num2))" ;;

3) echo "$num1 \* $num2 = $((num1 \* num2))" ;;

4) echo "$num1 / $num2 = $((num1 / num2))" ;;

\*) echo "Invalid choice" ;;

Esac

### To print days of a week.

#!/bin/bash

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

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

echo $day

done

### To print starting 4 months having 31 days.

#!/bin/bash

months = ("January" "March" "May" "July")

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

echo "$month has 31 days"

done

### Using functions,

### To find given number is Amstrong number or not

#!/bin/bash

is\_armstrong() {

num=$1

temp=$num

sum=0

length=${#num}

while [ $temp -gt 0 ]; do

digit=$((temp % 10))

sum=$((sum + digit\*\*length))

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 for Armstrong check:"

read arm\_num

is\_armstrong $arm\_num

### To find whether a number is palindrome or not

is\_palindrome() {

num=$1

temp=$num

rev=0

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

}

echo "Enter a number for Palindrome check:"

read pal\_num

is\_palindrome $pal\_num

### To print Fibonacci series upto n terms

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 for Fibonacci series:"

read fib\_num

fibonacci $fib\_num

### To find given number is prime or composite

is\_prime() {

num=$1

if [ $num -le 1 ]; then

echo "$num is neither prime nor composite"

return

fi

for ((i=2; i<=num/2; i++)); do

if [ $((num % i)) -eq 0 ]; then

echo "$num is a composite number"

return

fi

done

echo "$num is a prime number"

}

echo "Enter a number for Prime check:"

read prime\_num

is\_prime $prime\_num

### To convert a given decimal number to binary equivalent

decimal\_to\_binary() {

num=$1

binary=""

while [ $num -gt 0 ]; do

binary=$((num % 2))$binary

num=$((num / 2))

done

echo "Binary equivalent: $binary"

}

echo "Enter a decimal number for binary conversion:"

read dec\_num

decimal\_to\_binary $dec\_num