Manshaa Kakani | BE: II Year | Information Technology (IT): A

Operating System Lab Assignment: III

Table of Contents

[ Aim of assignment 1](#_Toc194744003)

[ To Perform 1](#_Toc194744004)

[ To Submit 1](#_Toc194744005)

[1. To find Largest of Three Numbers 1](#_Toc194744006)

[2. To find a year is leap year or not. 1](#_Toc194744007)

[3. To input angles of a triangle and find out whether it is valid triangle or not. 2](#_Toc194744008)

[4. To check whether a character is alphabet, digit or special character. 2](#_Toc194744009)

[5. To calculate profit or loss 2](#_Toc194744010)

[6. To print all even and odd number from 1 to 10 3](#_Toc194744011)

[7. To print table of a given number 3](#_Toc194744012)

[8. To find factorial of a given integer 3](#_Toc194744013)

[9. To print sum of all even numbers from 1 to 10. 3](#_Toc194744014)

[10. To print sum of digit of any number. 4](#_Toc194744015)

[11. To make a basic calculator which performs addition, subtraction, multiplication, division 4](#_Toc194744016)

[12. To print days of a week. 5](#_Toc194744017)

[13. To print starting 4 months having 31 days. 5](#_Toc194744018)

[14. Using functions, 5](#_Toc194744019)

[a. To find given number is Amstrong number or not 5](#_Toc194744020)

[b. To find whether a number is palindrome or not 6](#_Toc194744021)

[c. To print Fibonacci series upto n terms 6](#_Toc194744022)

[d. To find given number is prime or composite 7](#_Toc194744023)

[e. To convert a given decimal number to binary equivalent 8](#_Toc194744024)

# Aim of assignment

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 a b c

if [ $a -gt $b ] && [ $a -gt $c ]; then

echo "The largest number is $a"

elif [ $b -gt $c ]; then

echo "The largest number is $b"

else

echo "The largest number is $c"

fi

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

#!/bin/bash

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

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

#!/bin/bash

echo "Enter three angles of the triangle:"

read a b c

if [ $(($a + $b + $c)) -eq 180 ]; then

echo "It is a valid triangle"

else

echo "It is not a valid 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 "It is an alphabet"

elif [[ "$char" =~ [0-9] ]]; then

echo "It is a digit"

else

echo "It is a special character"

fi

## To calculate profit or loss

#!/bin/bash

echo "Enter cost price and selling price:"

read cp sp

if [ $sp -gt $cp ]; then

echo "Profit = $(($sp - $cp))"

elif [ $sp -lt $cp ]; then

echo "Loss = $(($cp - $sp))"

else

echo "No profit, no loss"

fi

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

#!/bin/bash

echo "Even numbers:"

for ((i=2; i<=10; i+=2)); do

echo $i

done

echo "Odd numbers:"

for ((i=1; i<=10; i+=2)); do

echo $i

done

## To print table of a given number

echo "Enter a number:"

read num

echo "Multiplication table for $num:"

for ((i=1; i<=10; i++)); do

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

done

## To find factorial of a given integer

#!/bin/bash

echo "Enter a number:"

read num

factorial=1

for ((i=1; i<=num; i++)); do

factorial=$(($factorial \* $i))

done

echo "Factorial is $factorial"

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

#!/bin/bash

sum=0

for ((i=2; i<=10; i+=2)); do

sum=$(($sum + $i))

done

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

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

#!/bin/bash

echo "Enter two numbers:"

read a b

echo "Choose an 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 "Division by zero is not allowed"

fi

;;

\*) echo "Invalid operation";;

esac

## To print days of a week.

#!/bin/bash

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

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

echo $day

done

## To print starting 4 months having 31 days.

#!/bin/bash

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

echo "Months with 31 days:"

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

echo $month

done

## Using functions,

## To find given number is Amstrong number or not

#!/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

## To find whether a number is palindrome or not

#!/bin/bash

is\_palindrome() {

num=$1

reverse=$(echo $num | rev)

if [ $num -eq $reverse ]; then

echo "$num is a palindrome"

else

echo "$num is not a palindrome"

fi

}

echo "Enter a number:"

read n

is\_palindrome $n

## To print Fibonacci series upto n terms

#!/bin/bash

fibonacci() {

n=$1

a=0

b=1

echo "Fibonacci series:"

for ((i=0; i<n; i++)); do

echo -n "$a "

temp=$a

a=$b

b=$(($temp + $b))

done

echo

}

echo "Enter the number of terms:"

read n

fibonacci $n

## To find given number is prime or composite

#!/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)) -eq 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

## To convert a given decimal number to binary equivalent

#!/bin/bash

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:"

read n

decimal\_to\_binary $n