**Lab-Assignment 3**

* **Aim:** To create shell scripts for the following questions
* **To perform**: To code and solve the following
* **To submit:** Shell scripts for following

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

#! /bin/bash

echo "Enter three numbers:"

read a b c

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

echo "$a is the largest"

elf [ $b -ge $a ] && [ $b -ge $c ]; then

echo "$b is the largest"

else

echo "$c is the largest"

fi

**2. 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

**3. 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 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

**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 "Alphabet"

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

echo "Digit"

else

echo "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 -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

**6. 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 $i

fi

done

echo "Odd numbers from 1 to 10:"

for i in {1..10}; do

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

echo $i

fi

done

**7. 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

**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 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"

**10. 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"

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

#!/bin/bash

echo "Enter first number:"

read a

echo "Enter second number:"

read b

echo "Choose operation: + - \* /"

read op

case $op in

+) echo "$a + $b = $((a + b))" ;;

-) echo "$a - $b = $((a - b))" ;;

\\*) echo "$a \* $b = $((a \* b))" ;;

/) echo "$a / $b = $((a / b))" ;;

\*) echo "Invalid operation" ;;

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" "August" "October" "December")

echo "First 4 months with 31 days:"

count=0

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

echo "$month"

((count++))

if [ $count -eq 4 ]; then

break

fi

done

**14. Using functions,**

**a. To find given number is Amstrong number or not**

**b. To find whether a number is palindrome or not**

**c. To print Fibonacci series upto n terms**

**d. To find given number is prime or composite**

**e. To convert a given decimal number to binary equivalent**

#!/bin/bash

**is\_armstrong()** {

echo "Enter number:"

read num

n=$num

sum=0

while [ $n -gt 0 ]; do

digit=$((n % 10))

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

n=$((n / 10))

done

if [ $sum -eq $num ]; then

echo "Armstrong number"

else

echo "Not an Armstrong number"

fi

}

**is\_palindrome()** {

echo "Enter number:"

read num

n=$num

rev=0

while [ $n -gt 0 ]; do

digit=$((n % 10))

rev=$((rev \* 10 + digit))

n=$((n / 10))

done

if [ $rev -eq $num ]; then

echo "Palindrome"

else

echo "Not a palindrome"

fi

}

**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

}

**is\_prime()** {

echo "Enter number:"

read num

if [ $num -le 1 ]; then

echo "Not Prime"

return

fi

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

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

echo "Composite"

return

fi

done

echo "Prime"

}

**decimal\_to\_binary()** {

echo "Enter decimal number:"

read num

binary=""

while [ $num -gt 0 ]; do

rem=$((num % 2))

binary="$rem$binary"

num=$((num / 2))

done

echo "Binary: $binary"

}