OPERATING SYSTEM(4ITRC2)

IT IV(Semester)

Submitted by

Name of Student Kanishka Joshi

Roll Number – 23I4141

Branch with Section – IT(B)

Submitted to

MRS_JASNEET KAUR

Assistant Professor, Department of Information Technology

Institute of Engineering and Technology

Devi Ahilya Vishwavidhyalaya, Indore (M.P.) India

(www.iet.dauniv.ac.in**)**

Session JAN-APRIL,2025

1.To find Largest of Three Numbers?

```
#!/bin/bash

# Prompt user to enter three numbers
echo "Enter the first number:"
read num1
echo "Enter the second number:"
read num2
echo "Enter the third number:"
read num3

# Compare the numbers to find the largest
if [ $num1 -gt $num2 ] && [ $num1 -gt $num3 ]; then
echo "The largest number is: $num1"
elif [ $num2 -gt $num1 ] && [ $num2 -gt $num3 ]; then
echo "The largest number is: $num2"
else
echo "The largest number is: $num3"
fi
```

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

```
#!/bin/bash

# Prompt the user to enter a year
echo "Enter a year:"
read year
```

```
# Check if the year is divisible by 4 and either not divisible by 100 or divisible by 400
if [ $(($year % 4)) -eq 0 ]; then
if [ $(($year % 100)) -ne 0 ] || [ $(($year % 400)) -eq 0 ]; then
echo "$year is a leap year."
else
echo "$year is not a leap year."
fi
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

# Prompt the user to enter the three angles of the triangle
echo "Enter the first angle of the triangle:"
read angle1
echo "Enter the second angle of the triangle:"
read angle2
echo "Enter the third angle of the triangle:"
read angle3

# Calculate the sum of the angles
sum=$((angle1 + angle2 + angle3))

# Check if the sum of angles is equal to 180 and all angles are positive
```

```
if [ $sum -eq 180 ] && [ $angle1 -gt 0 ] && [ $angle2 -gt 0 ] && [ $angle3 -gt 0 ]; then
echo "The angles form a valid triangle."
else
echo "The angles do not form a valid triangle."
Fi
```

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

```
#!/bin/bash

# Prompt the user to enter a character
echo "Enter a character:"
read char

# Check if the input is a single character
if [ ${#char} -ne 1 ]; then
echo "Please enter only one character."
exit 1
fi

# Use a case statement to identify the character type
case $char in
[A-Za-z])
echo "The character is an alphabet."
;;
[0-9])
echo "The character is a digit."
;;
*)
echo "The character is a special character."
;;
Esac
```

5.To calculate profit or loss

```
#!/bin/bash
```

```
# Prompt the user to enter the cost price and selling price
echo "Enter the cost price:"
read cost_price
echo "Enter the selling price:"
read selling_price

# Calculate profit or loss
if [ $selling_price -gt $cost_price ]; then
profit=$(($selling_price - $cost_price))
echo "You made a profit of $profit."
elif [ $selling_price -lt $cost_price ]; then
loss=$(($cost_price - $selling_price))
echo "You incurred a loss of $loss."
else
echo "No profit, no loss."
Fi
```

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

```
#!/bin/bash

# Print even numbers from 1 to 10
echo "Even numbers from 1 to 10 are:"
for ((i=1; i<=10; i++)); do
if [ $(($i % 2)) -eq 0 ]; then
echo $i
fi
done

# Print odd numbers from 1 to 10
echo "Odd numbers from 1 to 10 are:"
for ((i=1; i<=10; i++)); do
if [ $(($i % 2)) -ne 0 ]; then
echo $i
fi
done
```

7. To print table of a given number

```
#!/bin/bash
```

```
# Prompt the user to enter a number
echo "Enter a number to print its multiplication table:"
read num

# Print the multiplication table
echo "Multiplication table for $num:"
for ((i=1; i<=10; i++)); do
result=$(($num * $i))
echo "$num x $i = $result"
done
```

8. To find factorial of a given integer

```
#!/bin/bash

# Prompt the user to enter a positive integer
echo "Enter a positive integer to find its factorial:"
read num

# Initialize factorial to 1
factorial=1

# Check if the number is valid
if [ $num -lt 0 ]; then
echo "Factorial is not defined for negative numbers."
exit 1
fi

# Calculate factorial using a loop
for ((i=1; i<=num; i++)); do
factorial=$(($factorial * $i))
done

# Display the result
echo "The factorial of $num is: $factorial"
```

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

```
#!/bin/bash

# Initialize sum to 0
sum=0

# Loop through numbers from 1 to 10
```

```
for ((i=1; i<=10; i++)); do
# Check if the number is even
if [ $(($i % 2)) -eq 0 ]; then
sum=$(($sum + $i))
fi
done

# Display the result
echo "The sum of all even numbers from 1 to 10 is: $sum"
```

10. To print sum of digit of any number.

```
#!/bin/bash

# Prompt the user to enter a number
echo "Enter a number to calculate the sum of its digits:"
read num

# Initialize sum to 0
sum=0

# Extract digits and calculate the sum
while [ $num -gt 0 ]; do
digit=$(($num % 10))
sum=$(($sum + $digit))
num=$(($num / 10))
done

# Display the result
echo "The sum of the digits is: $sum"
```

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

```
#!/bin/bash

# Function to perform calculations
calculate() {
case $operation in
1)
result=$(($num1 + $num2))
echo "The result of addition is: $result"
;;
```

```
2)
result=$(($num1 - $num2))
echo "The result of subtraction is: $result"
;;
3)
result=$(($num1 * $num2))
echo "The result of multiplication is: $result"
;;
4)
if [ $num2 -ne 0 ]; then
result=$(echo "scale=2; $num1 / $num2" | bc)
echo "The result of division is: $result"
else
echo "Division by zero is not allowed!"
fi
;;
*)
echo "Invalid operation selected!"
;;
esac
}

# Prompt the user to input two numbers
echo "Enter the first number:"
read num1
echo "Enter the second number:"
read num2

# Display menu for operations
echo "Select an operation to perform:"
echo "1. Addition"
echo "2. Subtraction"
echo "3. Multiplication"
echo "4. Division"
read operation

# Perform the selected calculation
Calculate
```

12. To print days of a week.

```
#!/bin/bash

# Define an array with the days of the week
```

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

# Loop through the array and print each day
echo "Days of the week are:"
for day in "${days[@]}"; do
echo $day
done
```

13. To print starting 4 months having 31 days.

```
#!/bin/bash

# Array of months with the number of days
months=("January" "March" "May" "July" "August" "October" "December")

# Print the first four months with 31 days
echo "The first 4 months with 31 days are:"
for ((i=0; i<4; i++)); do
echo ${months[$i]}
done
```

14. Using functions,

a. 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 num
is_armstrong $num
```

b. To find whether a number is palindrome or not

```
#!/bin/bash

is_palindrome() {
num=$1
reversed=0
temp=$num

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

if [ $reversed -eq $num ]; then
echo "$num is a palindrome."
else
echo "$num is not a palindrome."
fi
}

echo "Enter a number:"
read num
is_palindrome $num
```

c. To print Fibonacci series upto n terms

```
#!/bin/bash

fibonacci() {
n=$1
a=0
b=1
echo "Fibonacci series up to $n terms:"
echo $a
echo $b
for ((i=3; i<=n; i++)); do
c=$(($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

```
#!/bin/bash

is_prime() {
num=$1
if [ $num -le 1 ]; then
echo "$num is neither prime nor composite."
return
fi

for ((i=2; i*i<=num; i++)); do
if [ $(($num % $i)) -eq 0 ]; then
echo "$num is composite."
return
fi
done
echo "$num is prime."
}

echo "Enter a number:"
read num
is_prime $num
```

e. 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 num
decimal_to_binary $num
```