

Subject: Operating System	Subject Code:22516
Semester:5 th Semester	Course: Computer Engineering
Laboratory No: V118	Name of Subject Teacher: Prof. Natasha Brahme
Name of Student: Siddharth Shah	Roll Id: 22203A0041

Experiment No:	10
Title of Experiment	Execute shell script by using if statement

SETS QUESTIONS

1) Write shell script to display square of two numbers.



```
#!/bin/bash
echo "Enter the first number:"
read num1
echo "Enter the second number:"
read num2
square1=$((num1 * num1))
square2=$((num2 * num2))
echo "The square of $num1 is: $square1"
echo "The square of $num2 is: $square2"
```

Output:-

```
Enter the first number:
12
Enter the second number:
11
The square of 12 is: 144
The square of 11 is: 121
```

2) Write a shell script for menu driven program.



```
#!/bin/bash

while true;
do
    echo "1. Add two numbers"
    echo "2. Subtract two numbers"
    echo "3. Multiply two numbers"
    echo "4. Divide two numbers"
    echo "5. Exit"
    echo "Enter your choice:"

    read choice

    case $choice in
        1)
            echo "Enter the first number:"
            read num1
            echo "Enter the second number:"
            read num2
            result=$((num1 + num2))
            echo "The result of addition is: $result"
            ;;
        2)
            echo "Enter the first number:"
            read num1
            echo "Enter the second number:"
            read num2
            result=$((num1 - num2))
            echo "The result of subtraction is: $result"
            ;;
        3)
            echo "Enter the first number:"
            read num1
            echo "Enter the second number:"
            read num2
            result=$((num1 * num2))
            echo "The result of multiplication is: $result"
            ;;
```

```
4)
  echo "Enter the first number:"
  read num1
  echo "Enter the second number:"
  read num2
  if [ $num2 -ne 0 ]; then
    result=$((num1 / num2))
    echo "The result of division is: $result"
  else
    echo "Error: Division by zero is not allowed!"
  fi
  ;;
5)
  exit 0
  ;;
*)
  echo "Invalid choice!"
  ;;
esac
done
```

Output:-

```
1. Add two numbers
2. Subtract two numbers
3. Multiply two numbers
4. Divide two numbers
5. Exit
Enter your choice:
3
Enter the first number:
22
Enter the second number:
48
The result of multiplication is: 1056
1. Add two numbers
2. Subtract two numbers
3. Multiply two numbers
4. Divide two numbers
5. Exit
Enter your choice:
4
Enter the first number:
12
Enter the second number:
0
Error: Division by zero is not allowed!
1. Add two numbers
2. Subtract two numbers
3. Multiply two numbers
4. Divide two numbers
5. Exit
Enter your choice:
```

XII. PRACTICAL RELATED QUESTIONS.

1) Write and execute script for nested if statement.



```
#!/bin/bash

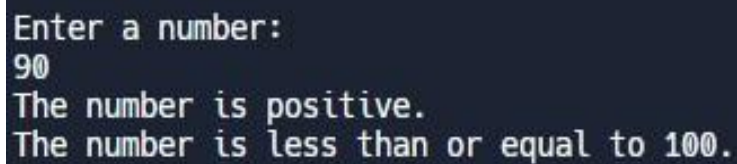
echo "Enter a number:"
read number

if [ $number -gt 0 ]; then
    echo "The number is positive."

    if [ $number -gt 100 ];
    then
        echo "The number is greater than 100."
    else
        echo "The number is less than or equal to 100."
    fi

elif [ $number -lt 0 ];
then
    echo "The number is negative."
else
    echo "The number is zero."
fi
```

Output:-



```
Enter a number:
90
The number is positive.
The number is less than or equal to 100.
```

2) Write difference between

- i. **if [condition]**
- ii. **if ((condition))**



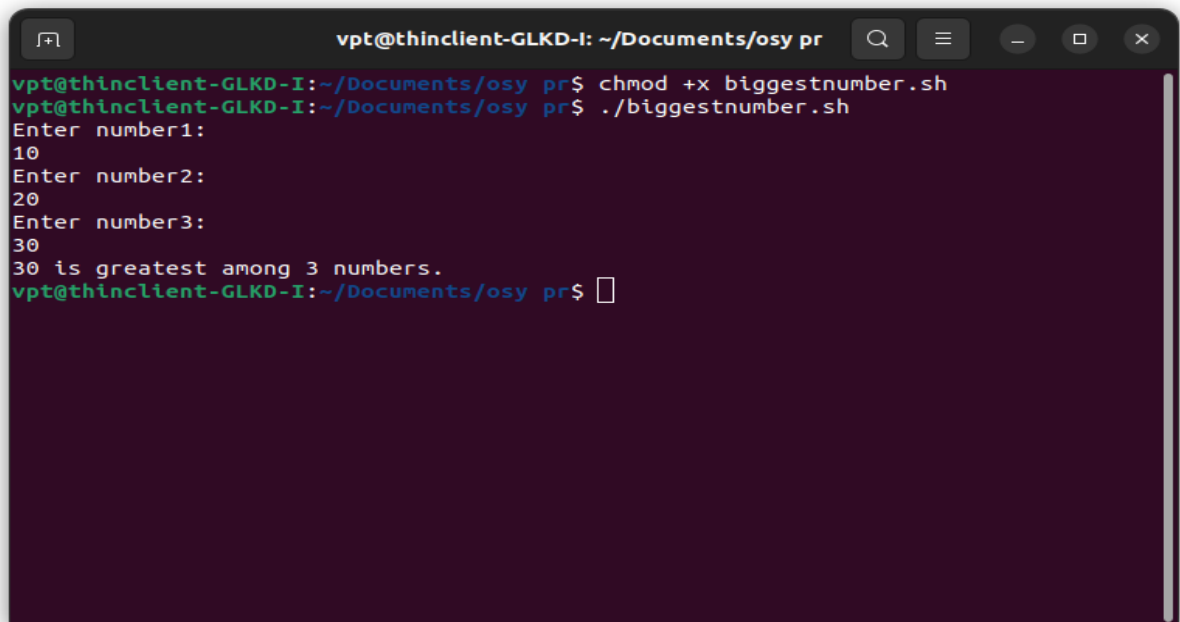
if[condition]	If((condition))
The single brackets [...] is the command	The double parentheses ((...)) is the format for bash arithmetic expansion.
It is used to create commands in statements.	It is used to test an arithmetic operation.
Syntax:- if [condition] then ### series of code fi	Syntax:- if ((condition)) then Statement goes here fi
Example:- if ["X" -lt "0"] then echo "X is less than zero" fi	Example:- if ((\$num -eq 42)) then echo "num is actually equal to 42" else echo "num is not equal to 42" fi

3) Write script for finding greatest number among the given numbers.



```
#!/bin/bash
echo "Enter number1: "
read num1
echo "Enter number2: "
read num2
echo "Enter number3: "
read num3
if [ "$num1" -ge "$num2" ] && [ "$num1" -ge "$num3" ]
then
echo "$num1 is greatest among 3 numbers."
elif [ "$num2" -ge "$num1" ] && [ "$num2" -ge "$num3" ]
then
echo "$num2 is greatest among 3 numbers."
else
echo "$num3 is greatest among 3 numbers."
fi
```

Output:-

A terminal window titled 'vpt@thinclient-GLKD-I: ~/Documents/osy pr' with search, menu, and window control icons. The terminal shows the execution of a script to find the greatest number among three inputs. The user enters 10, 20, and 30, and the script outputs that 30 is the greatest.

```
vpt@thinclient-GLKD-I:~/Documents/osy pr$ chmod +x biggestnumber.sh
vpt@thinclient-GLKD-I:~/Documents/osy pr$ ./biggestnumber.sh
Enter number1:
10
Enter number2:
20
Enter number3:
30
30 is greatest among 3 numbers.
vpt@thinclient-GLKD-I:~/Documents/osy pr$
```

XIII. EXERCISE.

1) Correct the following script and write its output.

- i. if [! -r "\$l"] then echo "File \$l is not readable-skipping."; fi
- ii. if ["\$X" -nt "/etc/passwd"]; then
echo "X is a file which is newer than/etc/passwd"
fi

→

- a) if [!-r"\$l"] then echo "File \$l is not readable-skipping."; fi

Error:-

```
utkarsha@utkarsha-VirtualBox:~/Desktop/Shell$ chmod ugo+x sam.sh
utkarsha@utkarsha-VirtualBox:~/Desktop/Shell$ ./sam.sh
File  is not readable - skipping.
utkarsha@utkarsha-VirtualBox:~/Desktop/Shell$
```

Correct code:

```
read l
if [! -r "$l"]
then
echo "File $l is not readable-skipping."
fi
```

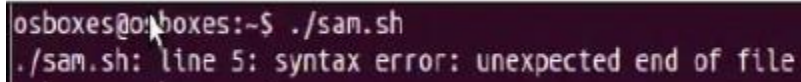
```
osboxes@osboxes:~$ ./sam.sh
ex.txt
"File ex.txt is not readable - skipping."
```



```
b)  if ["$X" -nt "/etc/passwd"]; then
      echo "X is a file which is newer than/etc/passwd"
    fi
```

ans:-

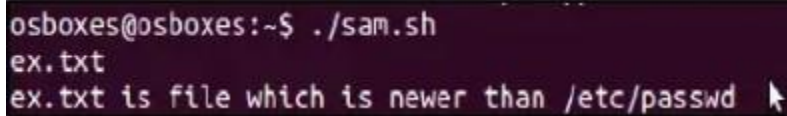
Error:-

A terminal window with a dark background. The prompt is 'osboxes@osboxes:~\$'. The user has entered './sam.sh'. The output is './sam.sh: line 5: syntax error: unexpected end of file'.

```
osboxes@osboxes:~$ ./sam.sh
./sam.sh: line 5: syntax error: unexpected end of file
```

Correct code:-

```
read x
if [$X -nt "/etc/passwd"]; then
echo "$X is a file which is newer than/etc/passwd"
fi
```

A terminal window with a dark background. The prompt is 'osboxes@osboxes:~\$'. The user has entered './sam.sh'. The output is 'ex.txt' followed by a new line and 'ex.txt is file which is newer than /etc/passwd'.

```
osboxes@osboxes:~$ ./sam.sh
ex.txt
ex.txt is file which is newer than /etc/passwd
```

CONCLUSION:

We successfully executed shell script by using if statement (Single decision, Double decision, Multiple if conditions)