**Experiment No:1**

**Aim:**

Introduction to Computer Hardware:- physical identification of major components of a computer system such as motherboard ,RAM,RAM modules ,Bus Slots , SMPS , Daughter Card , Internal Storage Devices, and Interfacing ports.

**CO1:**

Install and configure common operating system

**Procedure:**

MOTHER BOARD

A motherboard is a circuit board inside general-purpose computing systems,

including personal computers, smart televisions, smart monitors, and other similar

devices, which supports communication between different electrical components and

houses components such as the CPU, memory, etc.

• It is typically made of fiberglass and copper.

RAM

RAM stands for random access memory, and it’s one of the most fundamental elements of

computing. RAM is a temporary memory bank where your computer stores data it needs to

retrieve quickly. RAM keeps data easily accessible so your processor can quickly find it

without having to go into long-term storage to complete immediate processing tasks.

RAM MODULES

RAM modules, also known as RAM sticks or memory modules, are an integral part of a computer's memory system. RAM stands for Random Access Memory, and it serves as temporary storage for data that the computer's processor needs to access quickly. Unlike storage devices such as hard drives or SSDs, RAM is volatile memory, meaning that its contents are lost when the computer is powered off.

There are different types and form factors of RAM modules, including:

1. DIMM (Dual In-Line Memory Module)
2. SODIMM (Small Outline Dual In-Line Memory Module)

DAUGHTER CARD

A daughterboard is type of circuit board that plugs in or is attached to the motherboard or similar expansion card to extend its features and services. Like a motherboard, a daughterboard has sockets, pins, plugs and connectors to be attached to other boards.A daughterboard complements the existing functionality of a motherboard or an expansion card. A daughterboard is also known as daughter card, piggyback board, riser card or mezzanine board.

BUS SLOTS

Bus slots, also known as expansion slots, are physical connectors on a computer's motherboard that allow for the installation of expansion cards or add-on components. These slots provide a means to extend the capabilities of a computer by adding new functionality or upgrading existing components.

Here are some common types of bus slots:

PCI (Peripheral Component Interconnect)

PCI Express (PCIe)

AGP (Accelerated Graphics Port)

RAM Slots

SMPS

A switched-mode power supply, sometimes known as a switch-mode power supply or 'SMPS', is an electronic power supply that integrates a switching regulator for efficient electrical power conversion. Like other supplies, an SMPS transfers power from a DC or AC source to DC loads while converting voltage and current.

INTERNAL STORAGE DEVICES

Internal storage devices refer to the physical components within a computer or electronic device that are used to store data persistently. These devices are typically located inside the computer or device and are not intended to be easily removable by the user. Internal storage is used to store the operating system, software applications, and user data.

There are several types of internal storage devices commonly used in computers and other electronic devices:

1. Hard Disk Drive (HDD)
2. Solid State Drive (SSD)
3. Hybrid Drives
4. eMMC and UFS
5. M.2 and PCIe SSDs

INTERFACING PORTS

Interfacing ports, also known as input/output (I/O) ports or connectors, are physical interfaces on electronic devices that allow for the connection of peripheral devices or the transfer of data and signals. These ports provide a means for devices to communicate with each other, exchange information, and interact with external devices or networks.

Here are some commonly used interfacing ports:

1. USB (Universal Serial Bus)
2. HDMI (High-Definition Multimedia Interface)
3. DisplayPort
4. Ethernet
5. Audio Jacks
6. Thunderbolt
7. Serial and Parallel Ports

**Result:** The program is executed successfully and the output is verified

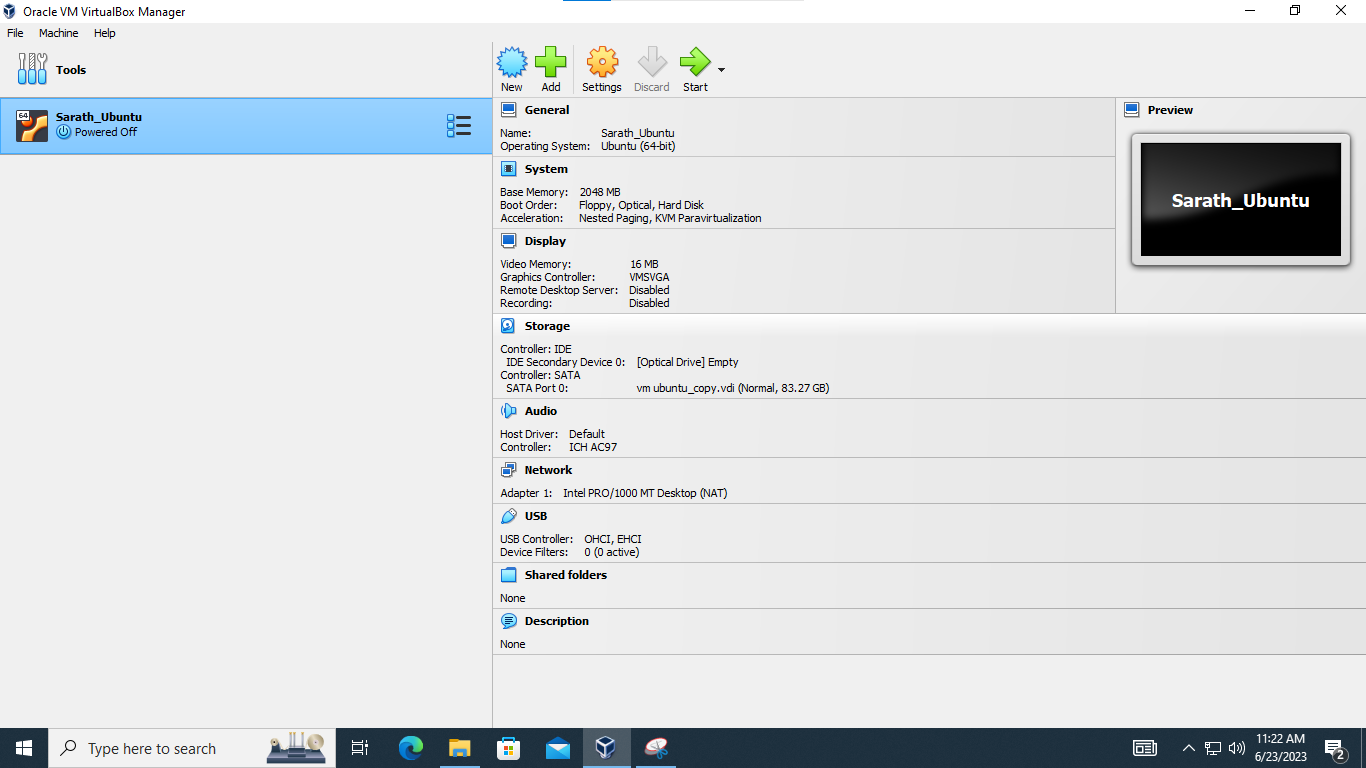
**Experiment No:2**

**Aim:** Install latest version of ubuntu on a Virtual machine.

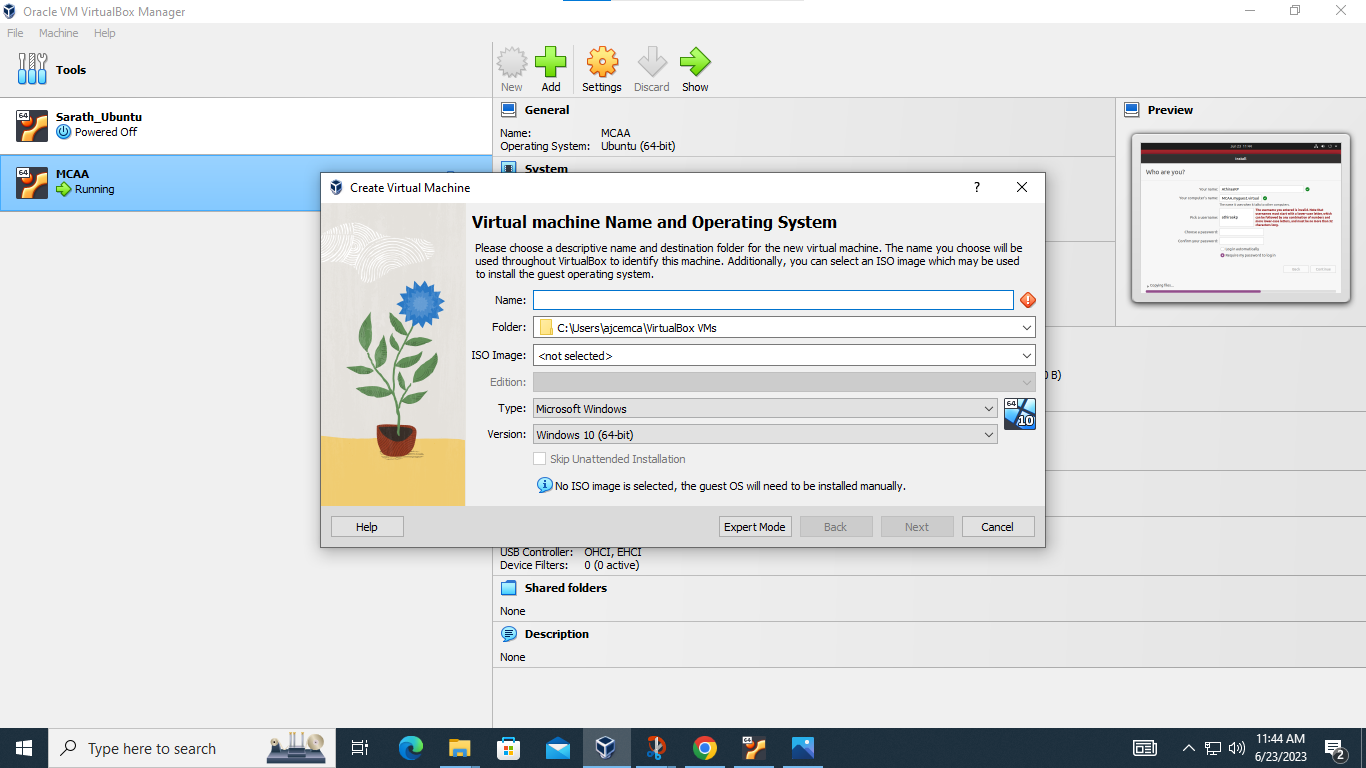
**CO:**

**Procedure:**

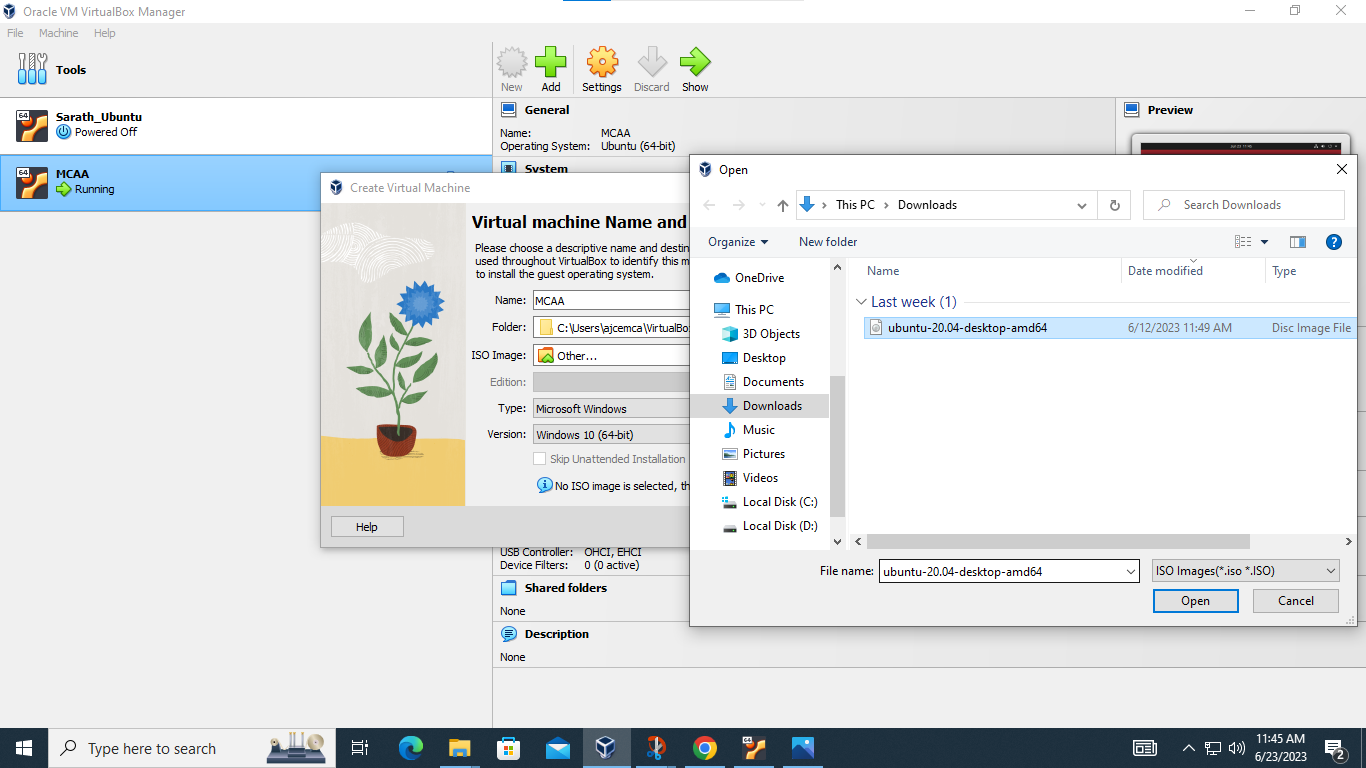
**Step 1:** Open Virtual box manager.

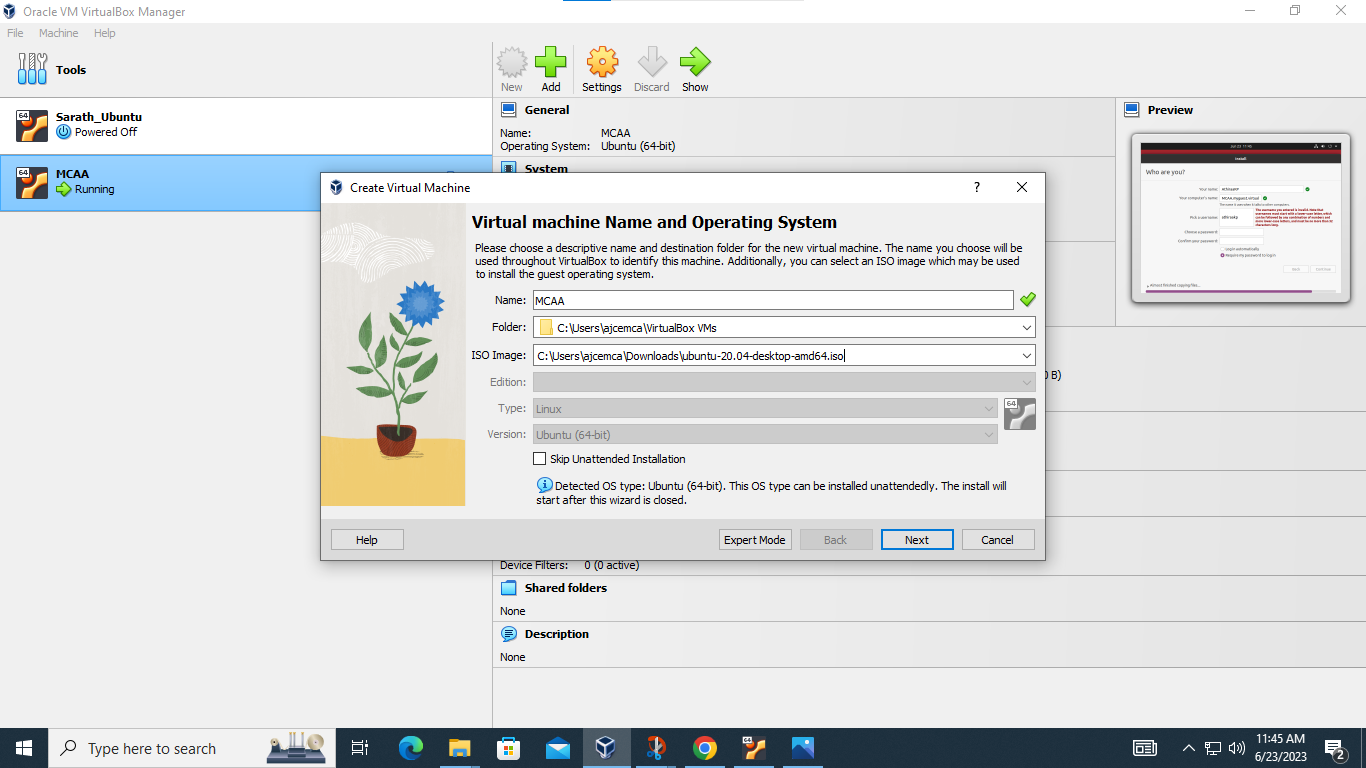
****

**Step 2:** Click new.

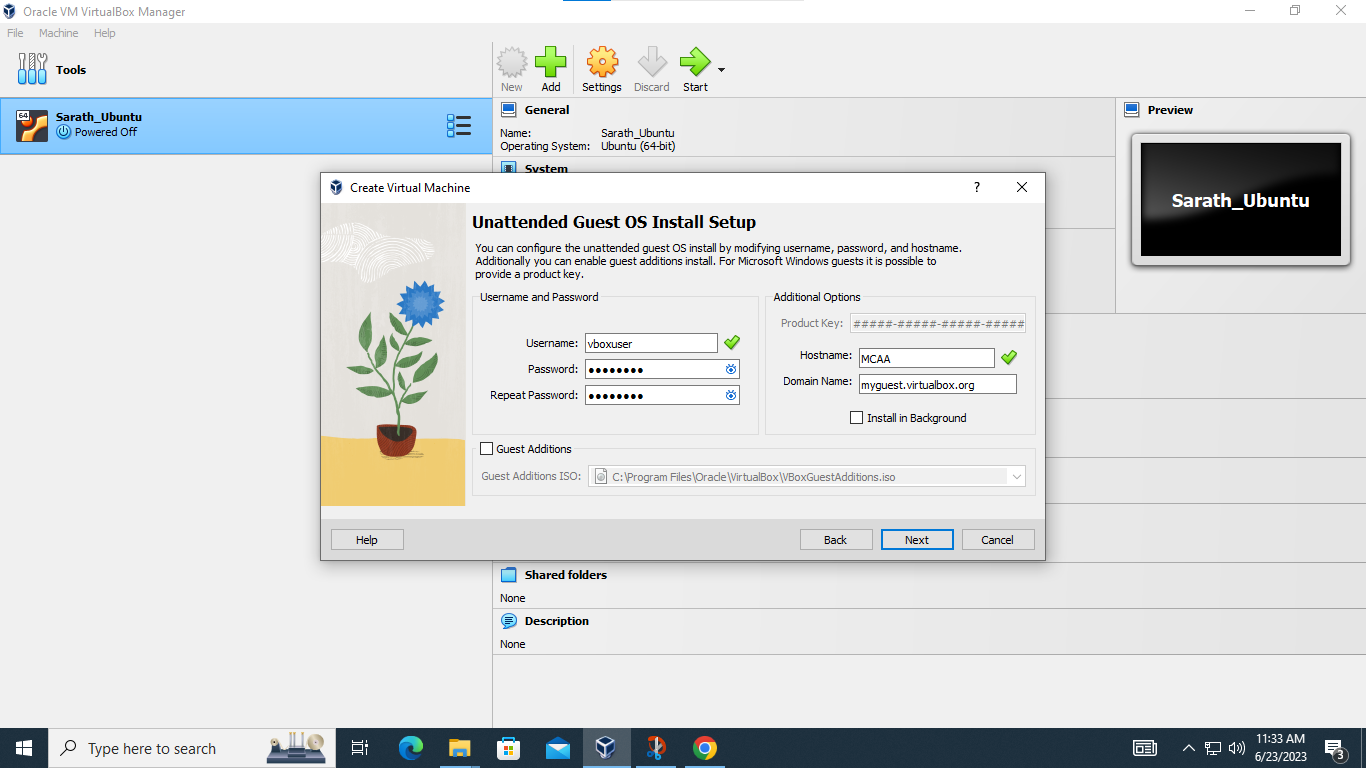
****

**Step 3:** Type a name for the virtual machine ,choose the iso file location, then click next.

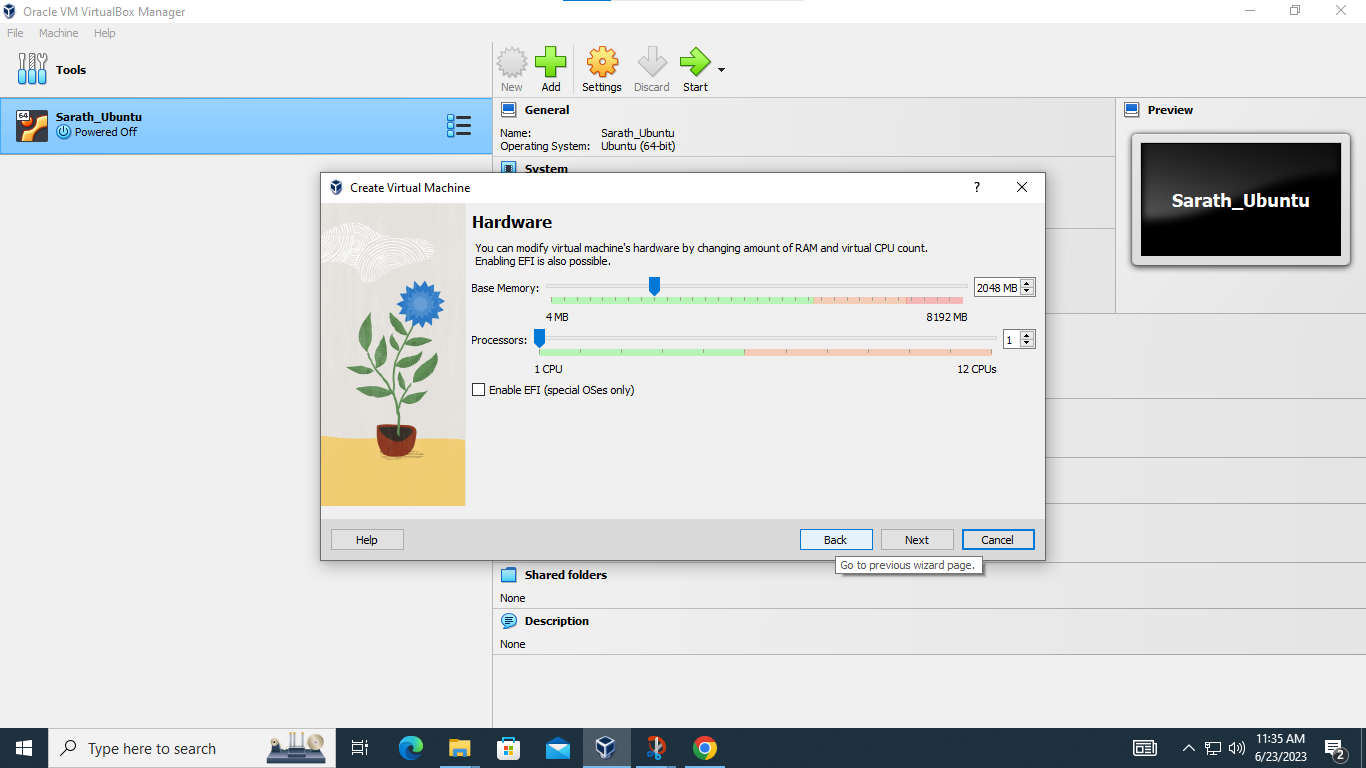
****

****

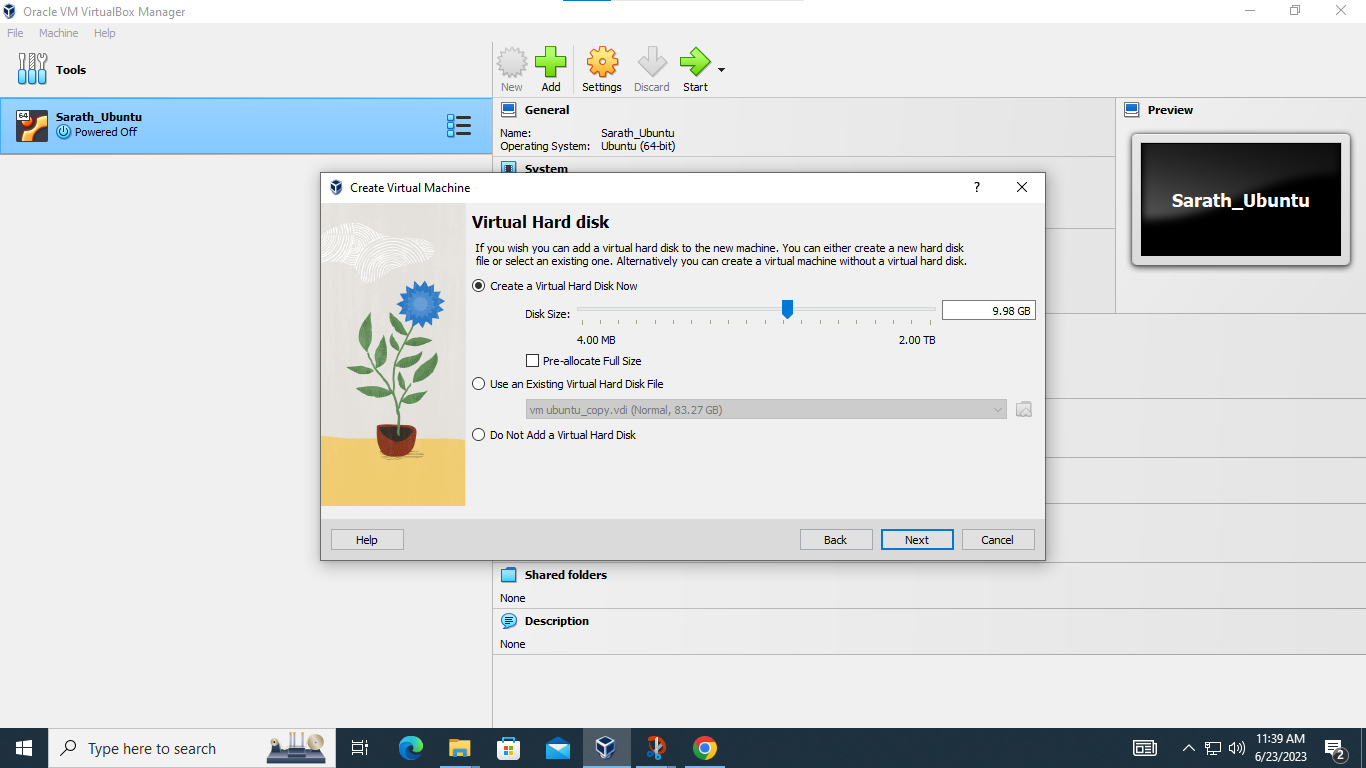
**Step 4:** Then a page opens to change username and password .Change it and click next

****

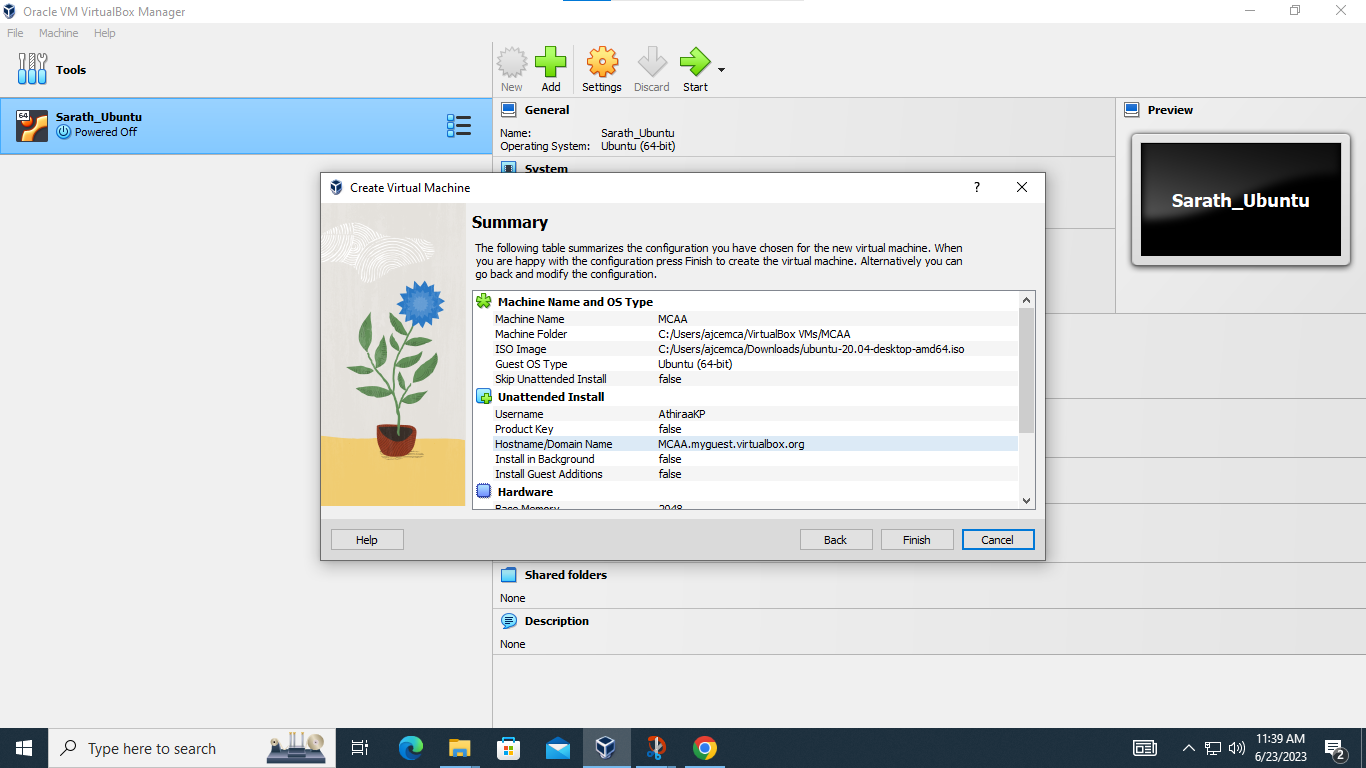
**Step 5:** Choose memory size and hardware size. Then click next.

****

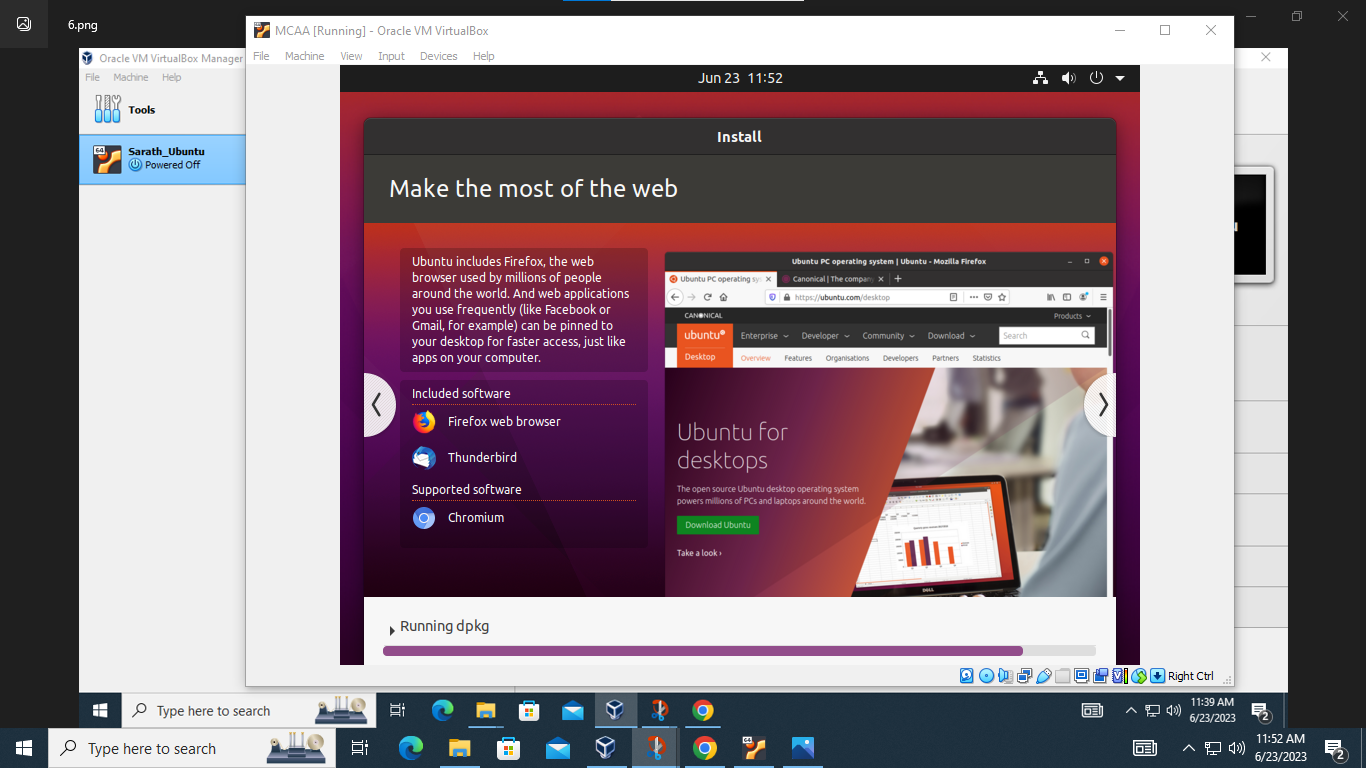
**Step 6:** Choose the Virtual Hard disk Size. Click next

****

**Step 7:** A preview page appears. Read it and click Finish

****

**Step 8:** Installation of latest version of Virtual Machine take place.

****

**Result:** The program is executed successfully and the output is verified

**Experiment No:3**

**Aim:** Familiarization of linux commands.

**CO 2:** Perform system administration.

**Procedure:**

**1) pwd :-** To identify the path of the working directory.

**$pwd**

**Output Screenshot:** 

**2) ls :-** To list the files and content.

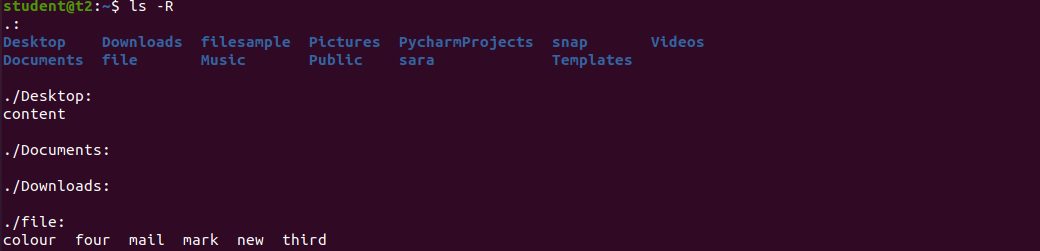
**$ls**

**Output Screenshot:**



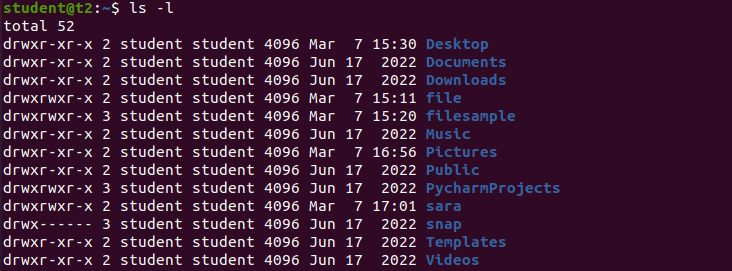
**2.1) ls -R :-** Tolist the contents of subdirectories.

**$ls -R**

**Output Screenshot:** 

**2.2) ls -l** **:-** Long listing of contents.

**$ls -l**

**Output Screenshot:** 

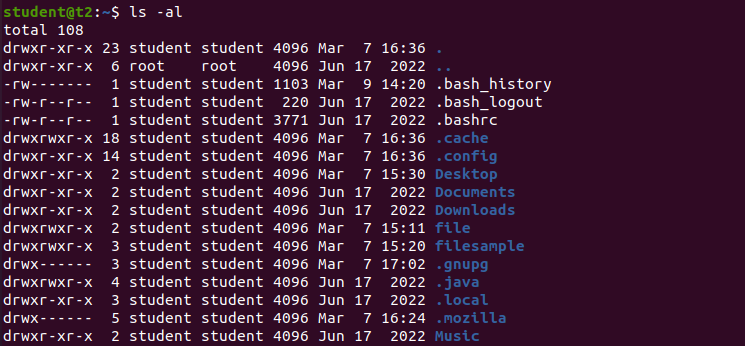
**2.3) ls -a :-** To view hidden files.

**$ls -a**

**Output Screenshot:**

**2.4) ls -al :-** We canlist files and dictionaries with detailed information.

**$ls -al**

**Output Screenshot:**

**2.5) ls -t :-** To list the files sorted in the order of last modified.

**$ls -t**

**Output Screenshot:** 

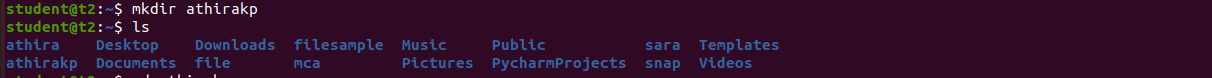
**2.6) ls -r :-**

**Output Screenshot:**



**3) mkdir :-** To create a new directory.

**$mkdir athirakp**

**Output Screenshot: **

**4) cd :-** To navigate through the directory.

**$cd athirakp**

**Output Screenshot:**

****

**4.1) cd -- :-** To move one step back to from the directory.

**$cd --**

**Output Screenshot: **

**5) history :-** To view the history and the commands which you have been executed for certain period of time.

**$history**

**Output Screenshot: **

**6) man :-** We can learn and understand about different commands right from shell.

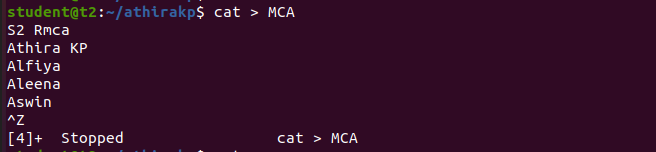
**$man ls**

**Output Screenshot:**

****

**7) cat > file\_name :-** To create a file.

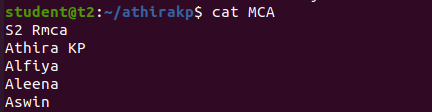
**$cat > MCA**

**Output Screenshot: **

**8) cat file\_name:-** To display contents in a file.

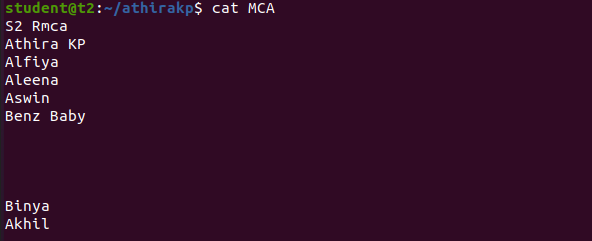
**$cat MCA**

**Output Screenshot:**



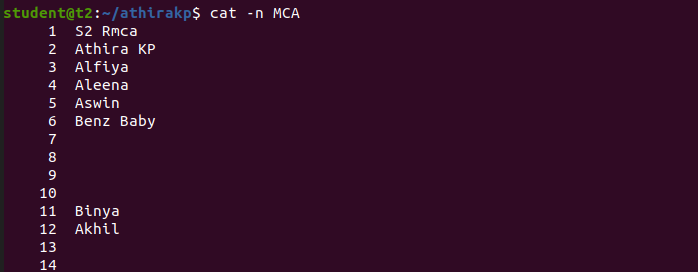
**8.1) cat > > file\_name:-** To append content to the existing file.

**$cat >> MCA**

**Output Screenshot: **

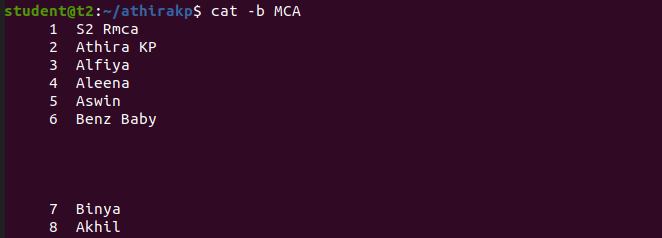
**8.2) cat -n file\_name :-** To display line with numbers

**$cat -n MCA**

**Output Screenshot: **

**8.3) cat -b file\_name :-** To remove numbering of empty lines

**$cat -b MCA**

**Output Screenshot: **

**Result:** The program is executed successfully and the output is verified

**Experiment No:4**

**Aim:** Familiarization of linux commands.

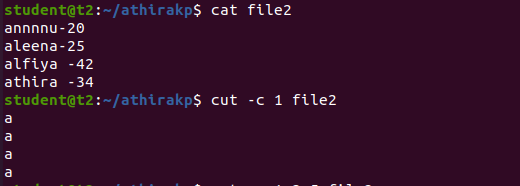
**CO2:** Perform system administration.

**Procedure:**

1. **cut -c :- To cut by character.**

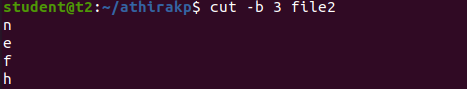
* 1. **cut -c filename :-** cut the specific character from the file.

**$cut -c 1 file2**

**Output Screenshot:**

* 1. **cut -b filename :-** cut the specific byte position from the file

**$cut -b 3 file2**

**Output Screenshot:**

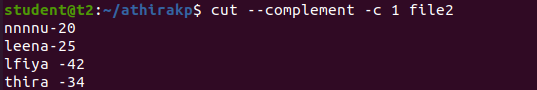
* 1. **cut -d filename :-** cut by columns and display one column using delimiter

**$cut -d - -f2 file2**

**Output Screenshot:**

* 1. **cut -- complement -c filename :-**To display all the content except cut content

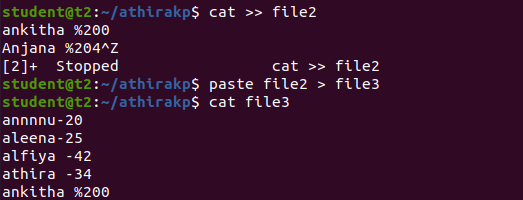
**$cut –complement -c 1 file2**

**Output Screenshot:**

1. **paste :-** To paste content from one file to another file.

**2.1) paste file1 > file2 :-** To Paste content from existing file to newly created file.

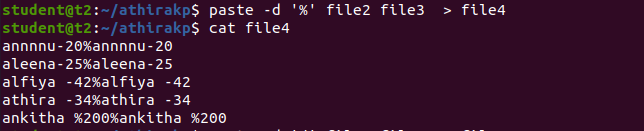
**$paste file2 >file3**

**Output Screenshot:**

**2.2) paste -d ‘%’ file2 file3 > file4 :-** To append the content in file2 and file3 to

newly created file file4.

$**paste -d ‘%’ file2 file3 > file4**

**Output Screenshot:**

**3) cp :-** To copy contents in a file to another file.

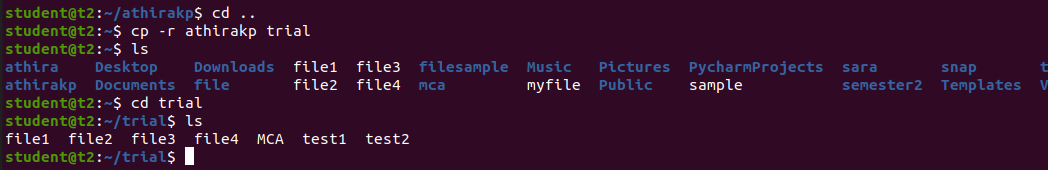
**3.1) cp test1 test2 :-** To copy the content in file test1 to test2.

**$cp test1 test2**

**Output Screenshot:**

**3.2) cp -r dir\_name trial :-** To copy files in a directory to another directory.

**$cp -r athirakp trial**

**Output Screenshot:**

**Result:** The program is executed successfully and the output is verified

**Experiment No:5**

**Aim:** Familiarization of linux commands.

**CO2:** Perform system administration.

**Procedure:**

**1) read :-** To read content.

**$read**

**Output Screenshot:**

**1.1) read var1 var2 var3 :-**To read content to specific variables.

**$read var1 var2 var3**

**Output Screenshot:**

**1.2) read -p “ ” :-** To prompt the content from user.

**$read -p “Enter the City:”**

**Output Screenshot:**

**1.3) read -n 6 -p “ ” :-** To display the specified number of lines.

**$read -n 6 -p “Enter your name:”**

**Output Screenshot:**

****

**1.4) read -s -p “ ” :-** To display the content securely.

**$read -s -p “Enter the password:”**

**Output Screenshot:**

**2) wc :-** To display the number of lines,number of words,number of bytes and filename.

**$wc filetest.txt**

**Output Screenshot:**

**2.1) wc -l filename:-** To display the number of lines only.

**$wc -l filetest.txt**

**Output Screenshot:**

**2.2) wc -m filename :-** To display the number of characters.

**$wc -m filetest.txt**

**Output Screenshot:**

**2.3) wc -c filename :-** To display the number of bytes.

**$wc -c filetest.txt**

**Output Screenshot:**

****

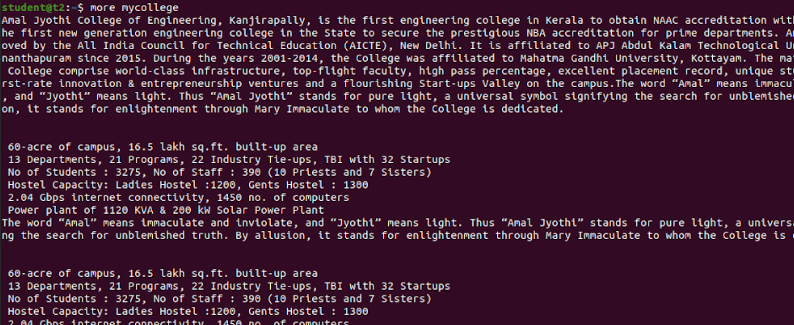
**2.4) wc -L filename :-** To display the length of longest word in the content.

**$wc -L filetest.txt**

**Output Screenshot:**

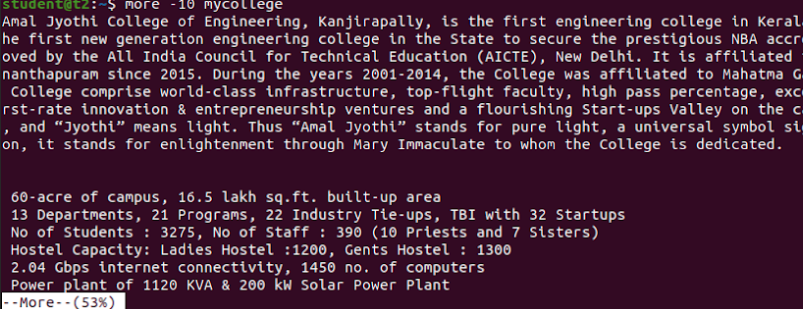
**3) more :-** To display the content like cat.The only difference is that in case of larger files cat command output will scroll full at a time.

**$more mycollege**

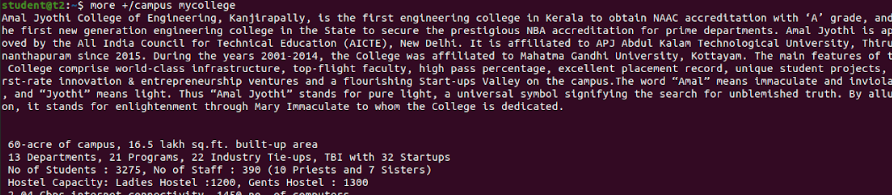
**Output Screenshot:**

**3.1) more -10 filename :-** Display the content after the specified no.of lines.

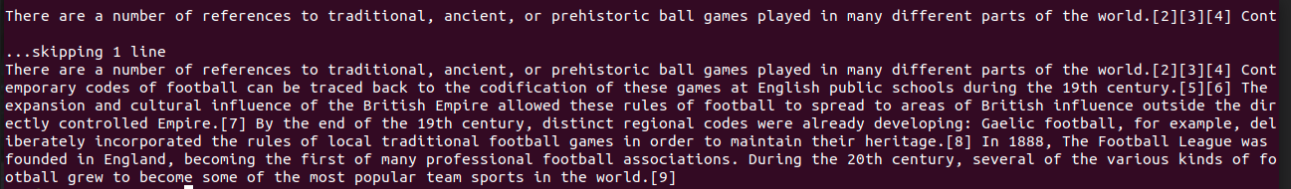
**$more -10 mycollege**

**Output Screenshot:**

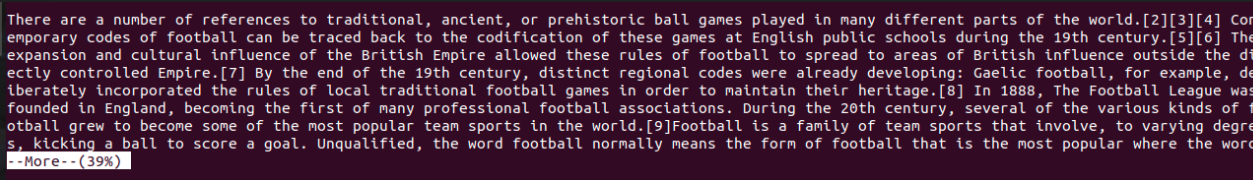
**3.2) more +/ campus filename :-** Display the content belonging the specified word.

**Output Screenshot:**

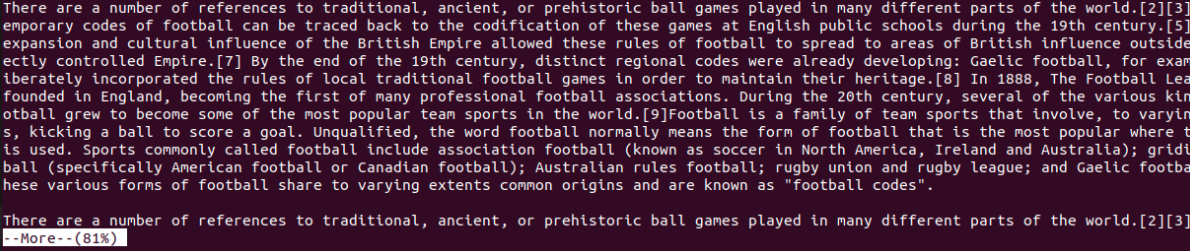
**3.3) ctrl+f :-** Moves one window forward.

**Output Screenshot:**

**3.4) ctrl+b :-** Moves one window backward.

**Output Screenshot:**

**3.5) ctrl+j :-** Moves one window forward by one line.

**Output Screenshot:**

**Result:** The program is executed successfully and the output is verified

**Experiment No:6**

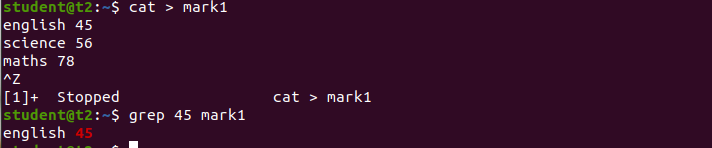
**Aim:** Familiarization of linux commands.

**CO2:** Perform system administration.

**Procedure**

**1) grep (global regular expression print) :-** Used to filter the content of a file which makes our search easy.

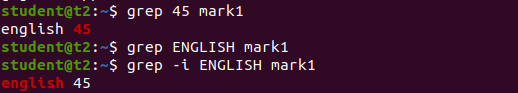
**$grep 45 mark1**

**Output Screenshot:**

**1.1) grep -i :-** To display search content.

**$grep -i English mark1**

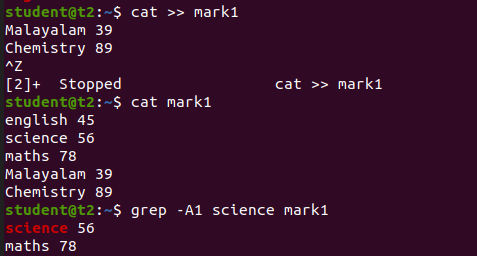
**Output Screenshot:**



**1.2) grep -A1 :-** To display one line after the search content.

**$grep -A1 science mark1**

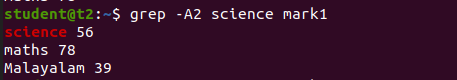
**Output Screenshot:**

****

**1.3) grep -A2 :-** To display 3 line after the search content.

**$grep -A2 science mark1**

**Output Screenshot:**



**1.4) grep -B2 :-**To display 2 lines before the search content.

**$grep -B2 science mark1**

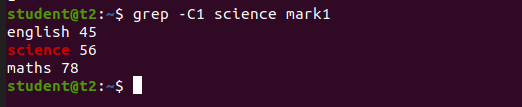
**Output Screenshot:**



**1.5) grep -C1 :-** To display one content before and after the search content.

**$grep -C1 science mark1**

**Output Screenshot:**



**1.6) grep -v :-** To display all the content except the search content.

**$grep -v Malayalam mark1**

**Output Screenshot:**

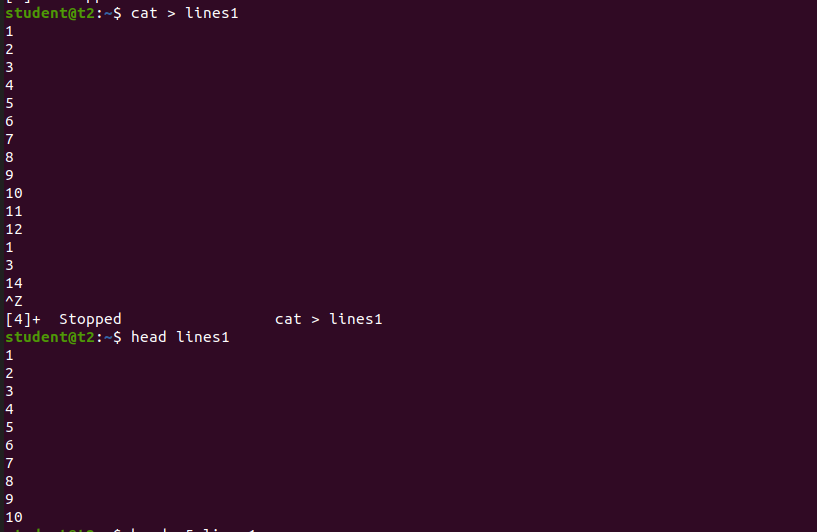
**2) cat filename|grep 5 :-** To highlight the search content.

**$ cat mark1|grep 9**

**Output Screenshot:**

**3) head filename :-** To display first 10 lines.

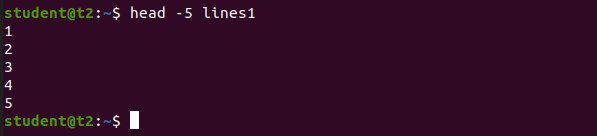
**$head lines1**

**Output Screenshot:**

**3.1) head -5 filename :-** To display specified lines from top.

$head -5 lines1

**Output Screenshot:**



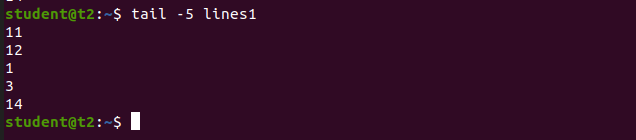
**4) tail filename :-** To display last 10 lines.

**$tail lines1**

**Output Screenshot:**

**4.1) tail -5 filename :-** To display specified lines from bottom.

**$tail -5 lines1**

**Output Screenshot:**

**Result:** The program is executed successfully and the output is verified

**Experiment No:7**

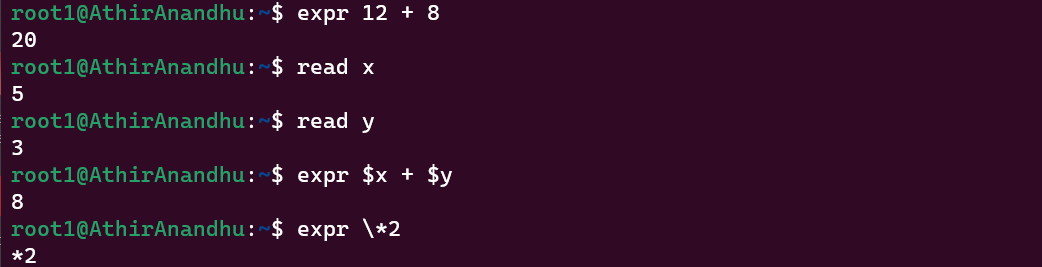
**Aim:** Familiarization of linux commands.

**CO2:** Perform system administration.

**Procedure:**

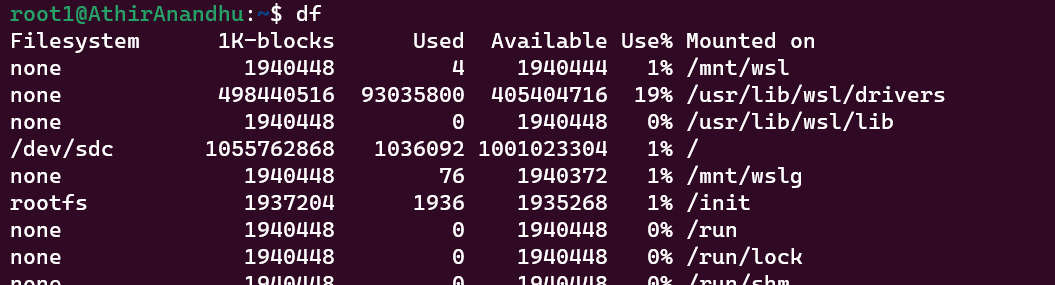
**1) expr :-** To evaluate the given expression and display output.

**$expr $x + $y**

**Output Screenshot:**

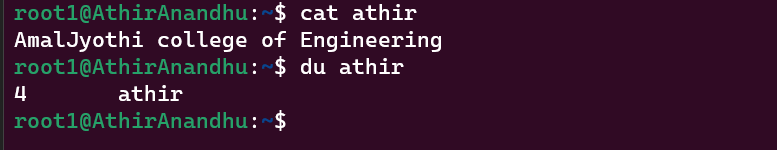
**2) df :-** Gives information about disk utilization.

**$df**

**Output Screenshot:**

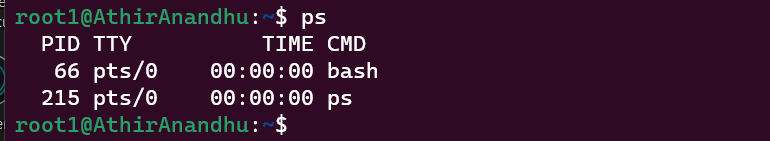
**3) du filename :-** To display how much space is utilized by a file.

**$du file2.txt**

**Output Screenshot:** ****

**4) ps :-** To check currently running process. Each process is identified by unique process id.

**$ps**

**Output Screenshot:**

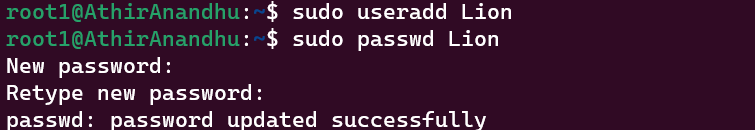
**4.1) ps -u :-** To identify currently running process by a particular user.

**$ps -u**

**Output Screenshot:**

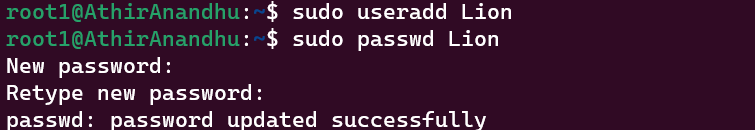
**5) sudo useradd username :-** To add or create a user.

**$sudo useradd Lion**

**Output Screenshot:**

**6) sudo passwd username :-** To update or change the password.

**$sudo passwd Lion**

**Output Screenshot:**

**7) sudo groupadd -g 1234 groupname :-** To create a group

**$sudo groupadd -g 1234 Animals**

**Output Screenshot:**

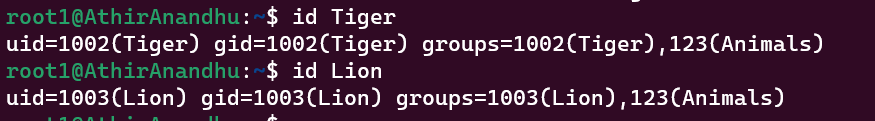
**8) sudo usermod -G groupname username :-** To assign a user to the group.

**$sudo usermod -G groupname Lion**

**Output Screenshot:**

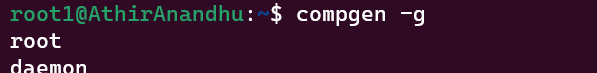
**9) id user :-** To view the information of the user.

**$id Lion**

**Output Screenshot:**

**10) compgen -g :-** To list the groups created.

**$compgen -g**

**Output Screenshot:**

**Result:** The program is executed successfully and the output is verified.

**Experiment No:8**

**Aim:** Familiarization of linux commands.

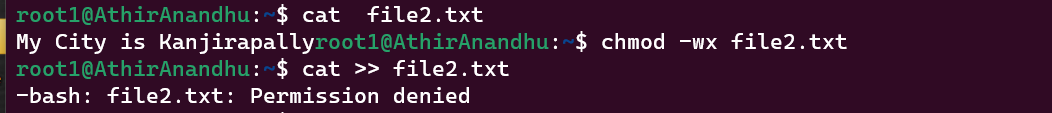
**CO2:** Perform system administration.

**Procedure:**

**1) chmod :-**  (change mode) chmod is used to change access permission of files and directories.

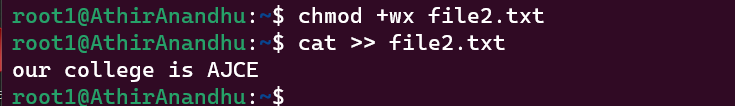
**1.1) chmod -wx filename :-** To deny the permission to write and execute.

**$chmod -wx file2.txt**

**Output Screenshot:**

**1.2) chmod +wx filename :-** To allow permission.

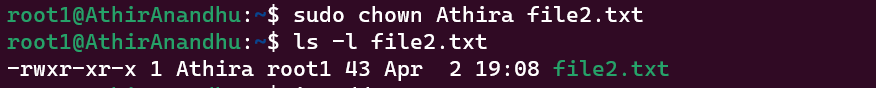
**$chmod +wx file2.txt 2**

**Output Screenshot:**

**2) sudo chown user filename :- (change owner)** To change ownership of file or a directory.

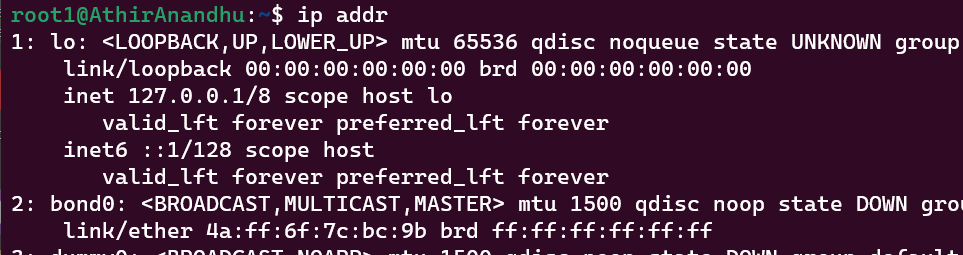
**$sudo chown Athira file2.txt**

**$ls -l file2.txt**

**Output Screenshot:**

**3) ip addr :-** To identify the ip address.

**$ip addr**

**Output Screenshot:**

**4) ssh ipaddress :- (Secure shell)**The protocol to connect remote system or server securely. It transfers data in encrypted form between host and client.

**$ssh** [**mca@192.168.6.40**](mailto:mca@192.168.6.40) **:-**

**Output Screenshot:**

**5) ssh -keygen :-** To generate a key for an authentication purpose. It specify the file to save the key.

**$ssh -keygen**

**Output Screenshot:**

**Result:** The program is executed successfully and the output is verified.

**Experiment No:- 9**

**Aim:-**

Write a script to convert the contents of a given file from uppercase to lowercase and count the number of lines, words, and characters of the resultant file. Also display the resultant file in descending order

**CO4 :-**

Write Shell Scripts required for system administration

**Procedure :-**

**Output Screenshot :-**

**Result:** The program is executed successfully and the output is verified

**Experiment No:- 10**

**Aim:**

Write script to perform the basic math operation as: Addition, Subtraction, multiplication, and division.

**CO4 :**

Write Shell Scripts required for system administration

**Procedure :**

echo "Enter Two numbers : "

read num1

read num2

echo "Enter Choice :"

echo "1. Addition"

echo "2. Subtraction"

echo "3. Multiplication"

echo "4. Division"

read ch

case $ch in

1)res=`echo $num1 + $num2 | bc`;;

2)res=`echo $num1 - $num2 | bc`;;

3)res=`echo $num1 \\* $num2 | bc`;;

4)res=`echo $num1 / $num2 | bc`;;

esac

echo "Result : $res"

**Output Screenshot :-**



**Result:** The program is executed successfully and the output is verified

**Experiment No:- 11**

**Aim:**

Write shell script to show various system configuration like

a) Currently logged user and his long name

b) Current shell

c) Your home directory

d)Your operating system type

e) Your current path setting

f) Your current working directory

g) Show all available shell

**CO4 :**

Write Shell Scripts required for system administration.

**Procedure :-**

#!/bin/bash

echo "a) Currently logged user and his long name:"

whoami

echo "b) Current shell: $SHELL"

echo "c) Your home directory: $HOME"

echo "d) Your operating system type:"

uname -o

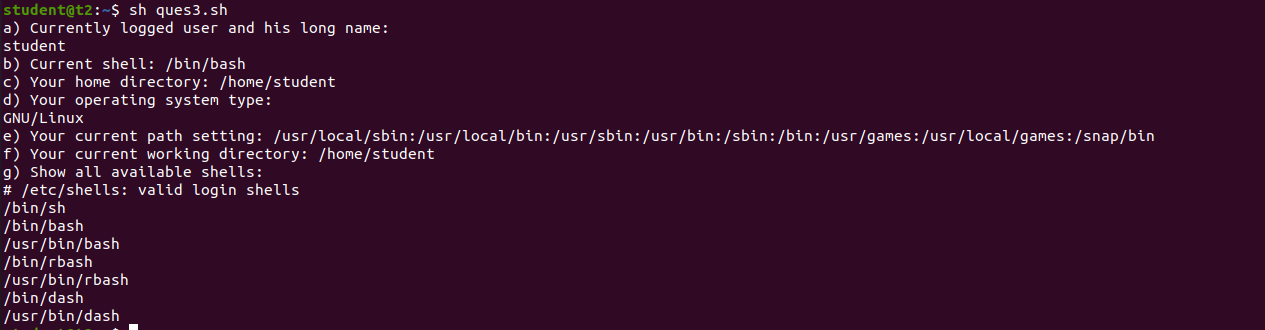
#cat /etc/os-release //details of os

echo "e) Your current path setting: $PATH"

echo "f) Your current working directory: $PWD"

echo "g) Show all available shells:"

cat /etc/shells

**Output Screenshot :-****Result:** The program is executed successfully and the output is verified

**Experiment No:- 12**

**Aim:**

Write a shell script to print the pattern.

1

2 2

3 3 3

4 4 4 4

**CO4 :**

Write Shell Scripts required for system administration.

**Procedure :-**

#!/bin/bash

for ((i=1;i<=5;i++))

do

for ((j=1;j<=i;j++))

do

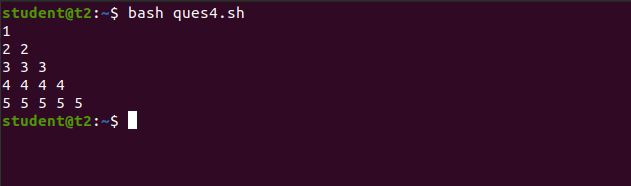
echo -e "$i \c"

done

echo " "

done

**Output Screenshot :-**



**Result:** The program is executed successfully and the output is verified

**Experiment No:- 13**

**Aim:**

Write a script to check whether the given string is palindrome or not.

**CO4 :**

Write Shell Scripts required for system administration.

**Procedure :-**

#!/bin/bash

echo "Enter String : "

read word

rev\_word=$(echo $word | rev)

if [ "$word" = "$rev\_word" ]; then

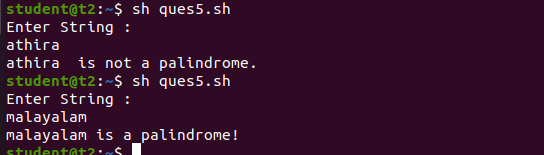
echo "$word is a palindrome!"

else

echo "$word is not a palindrome."

fi

**Output Screenshot :-**



**Result:** The program is executed successfully and the output is verified

**Experiment No:- 14**

**Aim:**

Write a shell script to find factorial of a given integer.

**CO4 :**

Write Shell Scripts required for system administration.

**Procedure :-**

#!/usr/bin/bash

echo "Enter an integer: "

read num

result=1

while [ $num -ge 1 ]

do

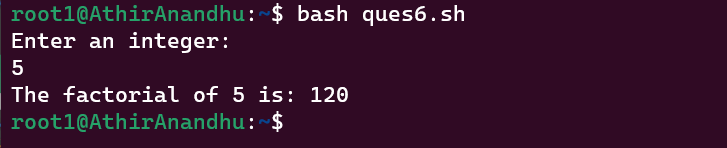
result=$((result \* num))

num=$((num - 1))

done

echo "The factorial of $num is: $result"

**Output Screenshot :**

****

**Result:** The program is executed successfully and the output is verified

**Experiment No:- 15**

**Aim:**

Execute the following scenario using basic Linux Commands

a. Create an untitled document myfile.txt using anyone editor.

b. Place the following text in myfile.txt and save it.

Neo: What are you trying to tell me? That I can dodge bullets?

Morpheus: No, Neo. I'm trying to tell you that when you're ready, you won't have to.

c. Count the number of characters, words, and lines in the file.

d. Find the occurrence of the word “tell” in the file.

e. Make two copies of myfile.txt with names myfile1.txt and myfile2.txt.

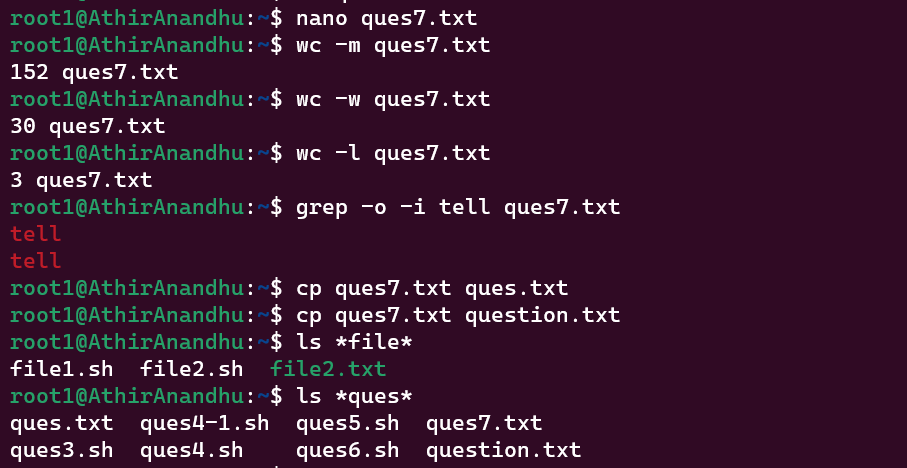
f. List all the filenames with the word file in the present working directory.

**CO4 :**

Write Shell Scripts required for system administration.

**Procedure :-**

**Output Screenshot :-**



**Result:** The program is executed successfully and the output is verified

**Experiment No:- 16**

**Aim:**

Execute the following scenario using basic Linux commands.

a) Login to your home directory

b) List contents of your current working directory

c) List all contents of your current working directory, including hidden files

d) Create a directory called April2023 inside your current working directory and go to

e) Create an empty file name file1

f) Make a copy of file1 to file2

g) Copy file1 as file2 in one directory up from the current directory

h) Clear the terminal window

**CO4 :-**

Write Shell Scripts required for system administration.

**Output Screenshot :-**

**Result:** The program is executed successfully and the output is verified

**Experiment No:- 17**

**Aim:**

Execute the following scenario using basic Linux commands.

a) Login to your home directory

b) Write the contents of syslog (/var/log/) on the screen a page at a time.

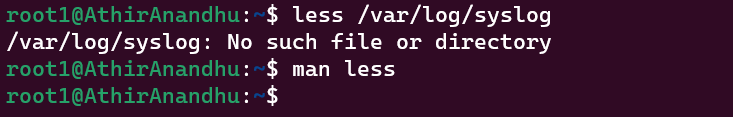
c) Read documentation on a command: less

**CO4 :**

Write Shell Scripts required for system administration.

**Procedure :-**

**Output Screenshot :-**



**Result:** The program is executed successfully and the output is verified

**Experiment No:- 18**

**Aim:**

Write a script to check whether the given number is a Armstrong or not.

**CO4 :**

Write Shell Scripts required for system administration.

**Procedure :-**

#!/bin/bash

echo "Enter a number: "

read c

x=$c

sum=0

r=0

n=0

while [ $x -gt 0 ]

do

r=`expr $x % 10`

n=`expr $r \\* $r \\* $r`

sum=`expr $sum + $n`

x=`expr $x / 10`

done

if [ $sum -eq $c ]

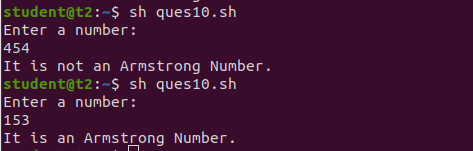
then

echo "It is an Armstrong Number."

else

echo "It is not an Armstrong Number."

fi

**Output Screenshot :-**

**Result:** The program is executed successfully and the output is verified

**Experiment No:19**

**Aim:** Familiarization of linux Networking commands.

**CO:**

**Procedure :-**

**1) if config :** To configure and display information about network interface**.**

**$ifconfig**

**Output Screenshot :-**

* 1. **sudo ifconfig enp5s0 down :-** To down the network

**$ sudo ifconfig enp5s0 down**

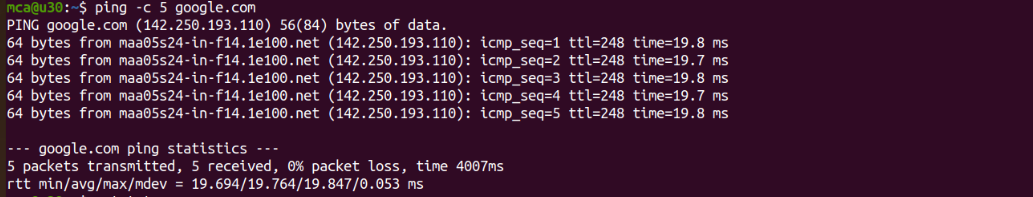
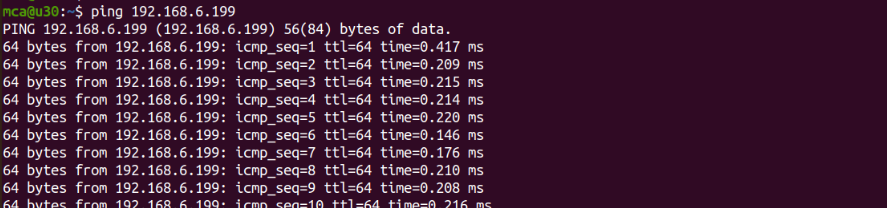
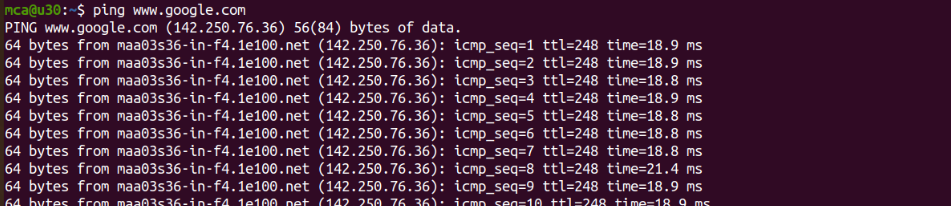
* 1. **sudo ifconfig enp5s0 up:-** To up the network

**$ sudo ifconfig enp5s0 up**

**Output Screenshot :-**

**2) ping :-** To test connectivity between host and server.

**$ping www.google.com**

**Output Screenshot :-**

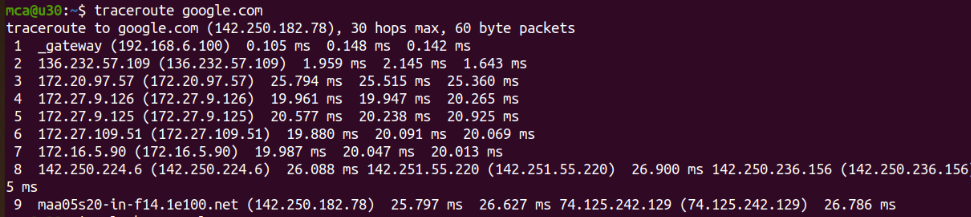
**3) whois :-** To find the information about registered domain.

**$whois www.google.com**

**Output Screenshot :-**

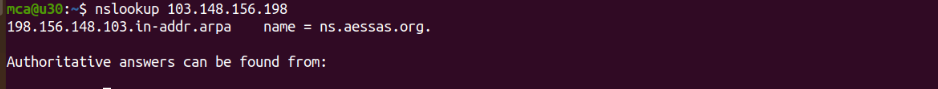
**4) traceroute :-** To identify the route taken by the packet through gateway to reach the destination.

**$traceroute google.com**

**Output Screenshot :-**

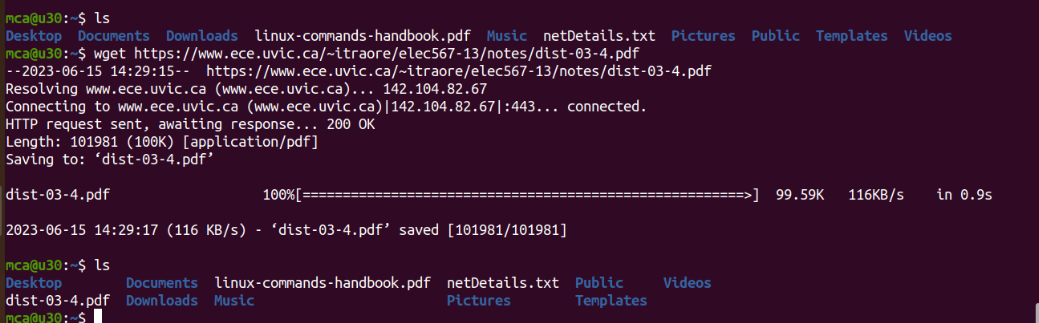
**5) nslookup :-** To identify the domain name system problems (DNS) and resolve it.

**$nslookup aesajce.com**

**Output Screenshot :-**

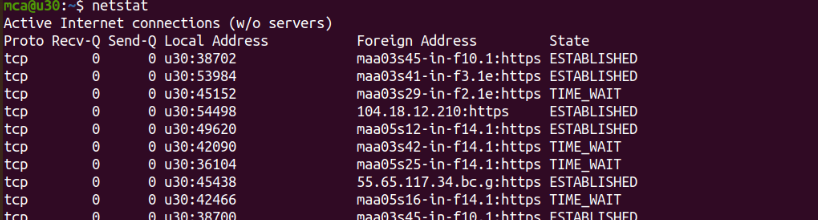
**6) wget :-** To retrieve a particular content from web browser.

**$wget http://www.uvic....**

**Output Screenshot :-**

**7) netstat :-** To display information about active network connections, listening ports, IP routing table and various network statistics.

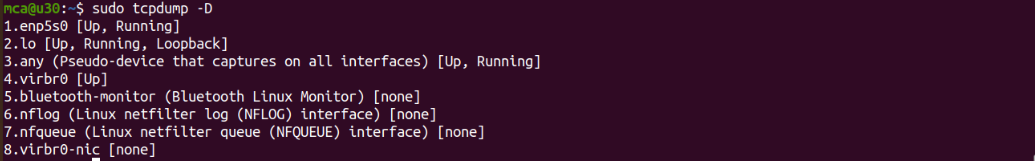
**$netstat**

**Output Screenshot :-**

**8) tcpdump :-** This will capture the packets from the current interface of the network through which the system is connected to the internet.

**$ sudo tcpdump**

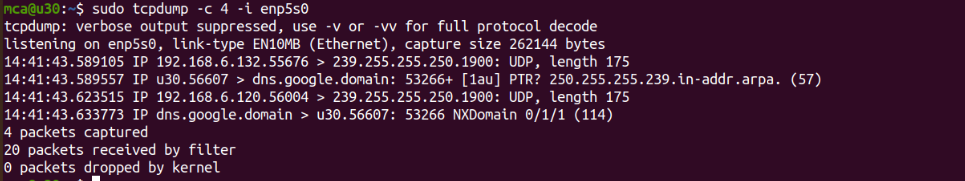
**Output Screenshot :-**

****

**8.1) sudo tcpdump -c -i enp5s0 :-** To capture particular number of packets from

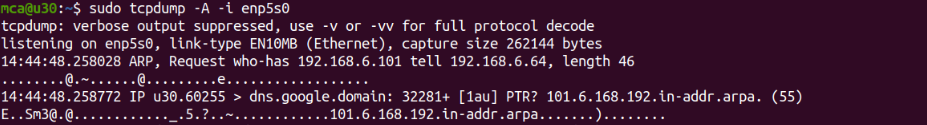
specified interface

**$ sudo tcpdump -c 4 -i enp5s0**

**Output Screenshot :-**

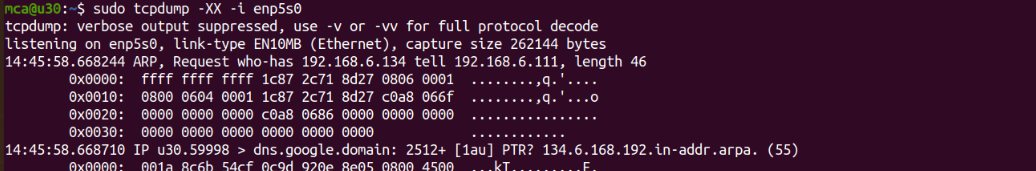
**8.2) sudo tcpdump -A -i enp5s0 :-** Print captured packet in ASCII format.

**$ sudo tcpdump -A -i enp5s0**

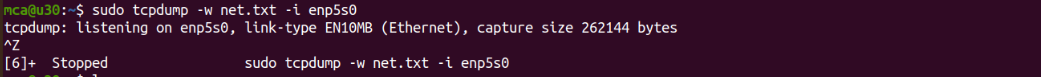
**Output Screenshot :-**

**8.3) sudo tcpdump -XX -i enp5s0 :-** Print captured packet in Hexadecimal format.

**$ sudo tcpdump -XX -i enp5s0**

**Output Screenshot :-**

**8.4) sudo tcpdump -w net.txt -i enp5s0 :-** To print captured packet in a text file.

**Output Screenshot :-**

**Result:** The program is executed successfully and the output is verified