



**MUTHOOT INSTITUTE OF TECHNOLOGY & SCIENCE**

**VARIKOLI P O, PUTHENCRUZ,**

**ERNAKULAM DISTRICT, PIN 682308**

**DEPARTMENT OF COMPUTER APPLICATIONS**

**MASTER OF COMPUTER APPLICATIONS**

**SEMESTER II**

**LAB RECORD**

**20MCA136 NETWORKING & SYSTEM  
ADMINISTRATION LAB**

**Name: AMAL TOM PARAKKADEN**

**Register Number: MUT23MCA-2010**

**Month: June Year: 2024**



## ***CERTIFICATE***

*Certified that this is the Bonafide Record of Practical work done in the  
Networking and System Administration Lab of Muthoot Institute of  
Technology and Science by Reg. No. **MUT23MCA-2010***

*Name: **AMAL TOM PARAKKADEN***

*for the partial fulfilment of the requirement for the award of the degree of  
Master of Computer Applications during the year **2023-24**.*

Head of the Department

Faculty in Charge

Dr. Saritha K

University Exam Reg. No: **MUT23MCA-2010** of 2024

Date of Examination: .....

Internal Examiner

External Examiner



### **Vision of Institute**

To be a centre of excellence for learning and research in engineering and technology, producing intellectually well-equipped and socially committed citizens possessing an ethical value system.

### **Mission of Institute**

- Offer well-balanced programme of instruction, practical exercise and opportunities in technology.
- Foster innovation and ideation of technological solutions on sustainable basis.
- Nurture a value system in students and engender in them a spirit of inquiry.

# INDEX

SL NO	PROGRAM DESCRIPTION	PAGE NO
1	Basic Linux Commands	1
2	Shell Programs	
	1.sum, product,quotient and difference of two numbers	14
	2.largest of two numbers	15
	3.largest among three numbers	16
	4.check a number is divisible by 3	17
	5. check a person is eligible to vote or not	18
	6.Sum , average and product of 4 integers.	19
	7.Factorial of a given number.	20
	8.Check number is prime or not.	21
	9.Simple calculator	22
	10.Accept a filename , starting and ending line numbers and display all the lines between the given number lines.	23
	11.Check if the input provided is a file or directory.	24
	12.Factorial of a number using while loop.	25
	13.Factorial of a number using until loop.	26
	14.Reverse a number	27
	15.Display prime numbers within a specific limit.	28
	16. Sum of digits of a number.	29
3	Introduction to Computer Hardware	30
4	Lamp Installation Procedure	37
5	Network Commands	42

## Basic Linux Commands

1. **pwd (Print Working Directory):** Use the pwd command to find out the path of the current working directory (folder) you're in. The command will return an absolute (full) path, which is basically a path of all the directories that starts with a forward slash (/). An example of an absolute path is /home/username.

```
mits@mits-Veriton-M200-H510:~/S2$ pwd
/home/mits/S2
```

2. **history :** When you have been using Linux for a certain period of time, you will quickly notice that you can run hundreds of commands every day. As such, running history command is particularly useful if you want to review the commands you have entered before.

```
mits@mits-Veriton-M200-H510:~/S2$ history
1  sudo apt update
2  sudo apt install build-essential
3  gedit
4  gcc helloworld.c
5  ./a.out
```

3. **man :**by using this command you can easily learn how to use.

```
NAME
  ls - list directory contents

SYNOPSIS
  ls [OPTION]... [FILE]...

DESCRIPTION
  List information about the FILES (the current directory by default). Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.
  Mandatory arguments to long options are mandatory for short options too.

  -a, --all
      do not ignore entries starting with .
```

4. **cd :**To navigate through the Linux files and directories, use the cd .It requires either the full path or the name of the directory, depending on the current working directory that you're in.

```
mits@mits-Veriton-M200-H510:~/S2$ cd MCA
mits@mits-Veriton-M200-H510:~/S2/MCA$
```

5. **ls:** The ls command is used to view the contents of a directory. By default, this command will display the contents of your current working directory. If you want to see the content of other directories, type ls and then the directory's path. There are variations you can use with the ls command:

- ls -R will list all the files in the sub-directories as well
- ls -l – long listing
- ls -a will show the hidden files
- ls -al will list the files and directories with detailed information like the permissions, size, owner, etc.
- ls -t lists files sorted in the order of “last modified”
- ls -r option will reverse the natural sorting order. Usually used in combination with
- other switches such as ls -tr. This will reverse the time-wise list string.

```
mits@mits-Veriton-M200-H510:~/S2$ ls
Aparna  MCA
mits@mits-Veriton-M200-H510:~/S2$ ls -R
.:
Aparna  MCA

./MCA:
d1

./MCA/d1:
d2

./MCA/d1/d2:
mits@mits-Veriton-M200-H510:~/S2$ ls -l
total 8
-rw-rw-r-- 1 mits mits   64 Feb 13 13:50 Aparna
drwxrwxr-x 3 mits mits 4096 Feb 13 14:16 MCA
```

6. **mkdir :** Use mkdir command to make a new directory — if you type mkdir Music it will create a directory called Music. To generate a new directory inside another directory, use this Linux basic command.

```
mits@mits-Veriton-M200-H510:~/S2$ mkdir networks
mits@mits-Veriton-M200-H510:~/S2$ cd networks
mits@mits-Veriton-M200-H510:~/S2/networks$
```

7. **rmdir**: If you need to delete a directory, use the rmdir command. However, rmdir only allows you to delete empty directories

```
mits@mits-Veriton-M200-H510:~/S2$ rmdir networks
mits@mits-Veriton-M200-H510:~/S2$ ls
Aparna MCA
```

8. **touch**: The touch command allows you to create a blank new file through the Linux command line.

```
mits@mits-Veriton-M200-H510:~/S2$ rmdir networks
mits@mits-Veriton-M200-H510:~/S2$ ls
Aparna MCA
```

9. **rm** : The rm command is used to delete directories and the contents within them. If you only want to delete the directory —as an alternative to rmdir — use rm -r. Be very careful with this command and double-check which directory you are in. This will delete everything and there is no undo. To remove a file use rm filename.

```
mits@mits-Veriton-M200-H510:~/S2$ rm a3
mits@mits-Veriton-M200-H510:~/S2$ ls
a1 a2 Aparna d1 MCA MITS
```

```
mits@mits-Veriton-M200-H510:~/S2$ rm -d emptydir
mits@mits-Veriton-M200-H510:~/S2$ ls
a1 a2 Aparna capitals d1 d2 India MCA MITS new New
```

- 10. cat:** cat (short for concatenate) is one of the most frequently used commands in Linux. It is used to list the contents of a file on the standard output stdout . To run this command, type cat followed by the file's name and its extension.

```
mits@mits-Veriton-M200-H510:~/S2$ cat>MITS
MITS
Muthoot institute of
technology
and
science
Puthencruz
varikoli
Ernakulam
^C
mits@mits-Veriton-M200-H510:~/S2$
```

- 11. echo:** echo command is used to move some data into a file. If you want to add the text, "Hello, my name is John" into a file called name.txt, you would type echo Hello, my name is John  
>> name.txt 2. head.

```
mits@mits-Veriton-M200-H510:~$ echo "Aparna"
Aparna
mits@mits-Veriton-M200-H510:~$ echo -e "\n\n aparna"

aparna
mits@mits-Veriton-M200-H510:~$
```

- 12. head:** The head command is used to view the first lines of any text file. By default, it will show the first ten lines, but you can change this number to your liking. If you only want to show the first five lines, type head -n 5 filename.txt.

```
mits@mits-Veriton-M200-H510:~/S2$ head -3 MITS
MITS
Muthoot institute of
technology
```



- 13. tail:** This one has a similar function to the head command, but instead of showing the first lines, the tail command will display the last ten lines of a text file. `tail -n filename.txt`.

```
mits@mits-Veriton-M200-H510:~/S2$ tail -3 MITS
Puthencruz
varikoli
Ernakulam
```

- 14. read:** read the contents of a line into a variable. The read command can be used with and without arguments. read command is used to read [options] [name...] . \$read \$read var1 var2 var3. \$echo &quot;[\$var1] [\$var2] [\$var3].

```
mits@mits-Veriton-M200-H510:~$ read name
Aparna
mits@mits-Veriton-M200-H510:~$ echo $name
Aparna
mits@mits-Veriton-M200-H510:~$
```

- 15. more:** Like cat command, more command displays the content of a file. Only difference is that, in case of larger files, &#39; cat&#39; command output will scroll off your screen while &#39; more&#39; command displays output one screenful at a time.

```
Muthoot Institute of Technology and Science (MITS) is promoted by Muthoot M. George Institute of Technology, a Section 25 Company within the Muthoot Group.

MITS is a self financing technical institution offering postgraduate and undergraduate engineering programmes, situated in the industrial suburb of Kochi, close to the Smart City and Info Park, approximately 15 kms from Vytilla Junction towards Muvattupuzha on the Cochin Madurai National Highway.

The promoters with their commanding presence in the domains of Financial Services, Hospitality, Healthcare, Renewable Energy and Information Technology, are committed to building MITS as a centre of excellence, focusing purely on merit. This "temple of learning" will provide to the students, state-of-the-art infrastructure, highly qualified and proficient teachers from various faculty of engineering and allied streams, embedded in a conducive environment where focus will be on their harmonious development.

The Muthoot Group, with a reputation that has been shaped over decades with high quality practices, total customer satisfaction and steady growth, spanning 130 years in the field of business, is a legacy built on God-given values of trust, truth, transparency and tradition and has become one of the top business houses today by the grace of God.

Muthoot takes its name from its Family, which is a branch of a traditional Ortho
--More-- (44%)
```

- 16. less:** The 'less' command is same as 'more' command but include some more features. It automatically adjusts with the width and height of the terminal window, while 'more' command cuts the content as the width of the terminal window get shorter.

```
Muthoot Institute of Technology and Science (MITS) is promoted by Muthoot M. George Institute of Technology, a Section 25 Company within the Muthoot Group.

MITS is a self financing technical institution offering postgraduate and undergraduate engineering programmes, situated in the industrial suburb of Kochi, close to the Smart City and Info Park, approximately 15 kms from Vytilla Junction towards Muvattupuzha on the Cochin Madurai National Highway.

The promoters with their commanding presence in the domains of Financial Services, Hospitality, Healthcare, Renewable Energy and Information Technology, are committed to building MITS as a centre of excellence, focusing purely on merit. This "temple of learning" will provide to the students, state-of-the-art infrastructure, highly qualified and proficient teachers from various faculty of engineering and allied streams, embedded in a conducive environment where focus will be on their harmonious development.

The Muthoot Group, with a reputation that has been shaped over decades with high quality practices, total customer satisfaction and steady growth, spanning 130 years in the field of business, is a legacy built on God-given values of trust, truth, transparency and tradition and has become one of the top business houses today by the grace of God.

Muthoot takes its name from its Family, which is a branch of a traditional Orthodox
AboutUs
```

- 17. cut :** The cut command is used for cutting out the sections from each line of files and writing the result to standard output. It can be used to cut parts of a line by byte position, character and file.

```
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ cut -c 1-3,6-8 states
kera
tam na
arucha
ass
bih
kartak
```

- 18. paste :** It is used to join files horizontally (parallel merging) by outputting lines consisting of lines from each file specified, separated by tab as delimiter, to the standard output. `paste [OPTION]... [FILES]...$ paste state.txt capital.txt.`

```
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ paste -d "|" number states capital
1|kerala|trivandrum
2|tamil nadu|chennai
3|arunachal pradesh|itanagar
4|assam|dispur
5|bihar|patna
6|karnataka|bangalore
```

- 19. uname :** The uname command, short for Unix Name, will print detailed information about your Linux system like the machine name, operating system, kernel, and so on  
\$uname, \$uname-r

```
mits@mits-Veriton-M200-H510:~$ uname -r
5.15.0-101-generic
mits@mits-Veriton-M200-H510:~$
```

- 20. cp :** cp command issued to copy files from the current directory to a different directory. For instance, the command cp scenery.jpg /home/username/Pictures would create a copy of scenery.jpg (from your current directory) into the Pictures directory. cp -i will ask for user's consent in case of a potential file overwrite. cp -p will preserve source files' mode, ownership and timestamp. cp -r will copy directories recursively. cp -u copies files only if the destination file is not existing or the source file is newer than the destination file.

```
mits@mits-Veriton-M200-H510:~/S2$ cp capitals d2
mits@mits-Veriton-M200-H510:~/S2$ cd d2
mits@mits-Veriton-M200-H510:~/S2/d2$ ls
capitals
```

- 21. mv :** The primary use of the mv command is to move files, it can also be used to rename files. The arguments in mv are similar to the cp command. You need to type mv, the file's name, and the destination's directory. mv file.txt /home/username/Documents .To rename files, the Linux is mv oldname.ext newname.ext

```
mits@mits-Veriton-M200-H510:~/S2$ ls
a1  Aparna      BasicLinuxCommands.docx  d1  File1  File3  India  MITS
a2  'assignment 1.docx' capitals                d2  File2  Fruits  MCA    new
mits@mits-Veriton-M200-H510:~/S2$ mv File1 File3
mits@mits-Veriton-M200-H510:~/S2$ ls
a1  Aparna      BasicLinuxCommands.docx  d1  File2  Fruits  MCA    new
a2  'assignment 1.docx' capitals                d2  File3  India   MITS   New
```

- 22. Find :** Similar to the locate command, using find also searches for files and directories. The difference is, you use the find command to locate files within a given directory. As an example, find /home/ -name notes.txt command will search for a file called notes.txt within the home directory and its subdirectories. Other variations when using the find are: To find files in the current directory use, find . -name notes.txt .To look for directories use, / -type d -name notes. txt.

```
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ find -name k*
./.thunderbird/yunpqhpm.default-release/key4.db
./s2mca/k1
./s2mca/k2
./s2mca/k3
./snap/firefox/common/.mozilla/firefox/2o3g1xgc.default/key4.
./.local/share/keyrings
```

- 23. grep :** Another basic Linux command that is undoubtedly helpful for everyday use is grep. It lets you search through all the text in a given file. To illustrate, grep blue notepad.txt will search for the word blue in the notepad file. Lines that contain the searched word will be displayed fully. Usually output of a previous command is piped into the grep command. For example, ls -l |grep "kernel".

```
mits@mits-Veriton-M200-H510:~/S2$ cat > India
India is my country
All Indians are my brothers and sisters.
I love India
Iam proud of INDIA
^C
mits@mits-Veriton-M200-H510:~/S2$ grep -i 'India' India
India is my country
All Indians are my brothers and sisters.
I love India
Iam proud of INDIA
```

```
mits@mits-Veriton-M200-H510:~/S2$ grep -c 'India' India
3
mits@mits-Veriton-M200-H510:~/S2$ grep -c -i 'India' India
4
mits@mits-Veriton-M200-H510:~/S2$ grep -v 'India' India
Iam proud of INDIA
```

- 24. df :** Use df command to get a report on the system's disk space usage, shown in percentage and KBs. If you want to see the report in megabytes, type df -m.

```
mits@mits-Veriton-M200-H510:~/S2$ df
Filesystem      1K-blocks      Used Available Use% Mounted on
udev             3916032         0    3916032   0% /dev
tmpfs             790912       2000     788912   1% /run
/dev/sda6       150080176 14998952 127384792  11% /
tmpfs            3954556         0    3954556   0% /dev/shm
tmpfs             5120          4         5116   1% /run/lock
```



- 25. du :** If you want to check how much space a file or a directory takes, the du (Disk Usage) command is the answer. However, the disk usage summary will show disk block numbers instead of the usual size format. If you want to see it in bytes, kilobytes, and megabytes, add the -h argument to the command line.

```
mits@mits-Veriton-M200-H510:~/S2$ du
4      ./MCA/d1/d2
8      ./MCA/d1
16     ./MCA
4      ./d2
4      ./d1/d2
12     ./d1
72     .
```

- 26. useradd :** This is available only to system admins .Since Linux is a multi- user system, this means more than one person can interact with the same system at the same time. useradd is used to create a new user, while passwd is adding a password to that user's account. To add a new person named John type, useradd John and then to add his password type, passwd 123456789.

```
mits@mits-Veriton-M200-H510:~/S2$ useradd mca
useradd: user 'mca' already exists
mits@mits-Veriton-M200-H510:~/S2$ sudo useradd mca
[sudo] password for mits:
Sorry, try again.
[sudo] password for mits:
Sorry, try again.
[sudo] password for mits:
useradd: user 'mca' already exists
```

```
mits@mits-Veriton-M200-H510:~/S2$ cat /etc/passwd|grep mca
mca:x:1004:1005:./home/mca:/bin/sh
mca2:x:1005:1006:./home/mca2:/bin/sh
```

- 27. userdel :**Remove a user is very similar to adding a new user. To delete the users account type, userdel UserName.

```
mits@mits-Veriton-M200-H510:~/S2$ sudo userdel mca2
mits@mits-Veriton-M200-H510:~/S2$ cat /etc/passwd|grep mca2
```

- 28. sudo :** Short for “SuperUser Do”, this command enables you to perform tasks that require administrative or root permissions. You must have sufficient permissions to use this command.

```
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ sudo
usage: sudo -h | -K | -k | -V
usage: sudo -v [-ABkns] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-ABkns] [-g group] [-h host] [-p prompt] [-U user] [-u user] [command]
usage: sudo [-ABbEHknPS] [-r role] [-t type] [-C num] [-D directory] [-g group] [-h host] [-p prompt] [-R directory] [-T timeout] [-u user] [VAR=value] [-i|-s]
[<command>]
usage: sudo -e [-ABkns] [-r role] [-t type] [-C num] [-D directory] [-g group] [-h host] [-p prompt] [-R directory] [-T timeout] [-u user] file ...
mits@mits-ThinkCentre-neo-50t-Gen-3:~$
```

- 29. passwd :** Changes passwords for user accounts. A normal user may only change the password for their own account, while the superuser may change the password for any account.

```
mits@mits-Veriton-M200-H510:~/S2$ cat /etc/passwd|grep mca
mca:x:1004:1005::/home/mca:/bin/sh
mca2:x:1005:1006::/home/mca2:/bin/sh
mits@mits-Veriton-M200-H510:~/S2$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
```

- 30. chmod :** To change directory permissions of file/ Directory in Linux. #chmod who what which file/directory chmod +rwx filename to add permissions. chmod -rwx directory name to remove permissions. chmod +x filename to allow executable permissions. chmod -wx filename totake out write and executable permissions. #chmod u+x test #chmod g- rwx test #chmod o-r test 4

```
total 784
-rw-rw-r-- 1 mca mits      0 Feb 13 14:30 a1
-rw-rw-r-- 1 mits mits      0 Feb 13 14:30 a2
-rw-rw-r-- 1 mits mits    64 Feb 13 13:50 Aparna
-rw-rw-r-- 1 mits mits  5056 Mar 11 15:47 'assignment 1.doc
-rw-rw-r-- 1 mits mits 497190 Mar  4 14:45 BasicLinuxComman
--wx-w--- 1 mits mits    63 Feb 15 09:16 capitals
drwxrwxr-x 3 mits mits   4096 Mar  4 15:40 d1
drwxrwxr-x 2 mits mits   4096 Apr  1 12:40 d2
-rw-rw-r-- 1 mits mits      0 Mar  4 15:17 File1
```

- 31. chown :** The chown command allows you to change the user and/or group ownership of a given file, directory. #chownTom Test

```
mits@mits-Veriton-M200-H510:~/S2$ sudo chown mca capitals
mits@mits-Veriton-M200-H510:~/S2$ ls -l
total 784
-rw-rw-r-- 1 mca mits      0 Feb 13 14:30 a1
-rw-rw-r-- 1 mits mits      0 Feb 13 14:30 a2
-rw-rw-r-- 1 mits mits    64 Feb 13 13:50 Aparna
-rw-rw-r-- 1 mits mits   5056 Mar 11 15:47 'assignment 1.docx'
-rw-rw-r-- 1 mits mits 497190 Mar  4 14:45 BasicLinuxCommands.docx
--wx-w---- 1 mca mits     63 Feb 15 09:16 capitals
drwxrwxr-x 3 mits mits   4096 Mar  4 15:40 d1
drwxrwxr-x 2 mits mits   4096 Apr  1 12:40 d2
-rw-rw-r-- 1 mits mits      0 Mar  4 15:17 File1
-rw-rw-r-- 1 mits mits      0 Mar  4 15:17 File2
-rw-rw-r-- 1 mits mits      0 Mar  4 15:17 File3
```

- 32. id :** id command in Linux is used to find out user and group names and numeric ID's (UID or group ID) of the current user.

```
mits@mits-Veriton-M200-H510:~$ id
uid=1000(mits) gid=1000(mits) groups=1000(mits),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),120(lpadmin),131(lxd),132(sambashare)
```

- 33. ps :** The ps command, short for Process Status, is a command line utility that is used to display or view information related to the processes running in a Linux system. PID – This is the unique process ID TTY– This is the type of terminal that the user is logged in to . TIME – This is the time in minutes and seconds that the process has been running .CMD – The command that launched the process

**Syntax:**

ps [options]

```
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ ps
  PID TTY          TIME CMD
 2212 pts/0        00:00:00 bash
 4844 pts/0        00:00:00 ps
mits@mits-ThinkCentre-neo-50t-Gen-3:~$
```

- 34. top :** top command is used to show the Linux processes. It provides a dynamic real-time view of the running system. Syntax: top [options]

```
mits@mits-Veriton-M200-H510:~/S2$ top -c
top - 13:29:00 up 1:21, 1 user, load average: 0.25, 0.40, 0.39
Tasks: 289 total, 1 running, 286 sleeping, 0 stopped, 2 zombie
%Cpu(s): 0.2 us, 0.0 sy, 0.0 ni, 99.8 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 7723.7 total, 1955.7 free, 2746.4 used, 3021.6 buff/cache
MiB Swap: 2048.0 total, 2048.0 free, 0.0 used, 3920.1 avail Mem

  PID USER      PR  NI   VIRT   RES   SHR S  %CPU  %MEM    TIME+  COMMAND
 2242 mits       20   0 4781928 717432 289476 S   1.3   9.1   8:40.01 /usr/lib/firefox/firefox
- new-window
 2522 mits       20   0 3084532 557540 124852 S   1.0   7.0   2:02.87 /usr/lib/firefox/firefox
- contentproc -childID 4 -isForBrowser -prefsLen 31655 -prefMapSize 238+
  762 avahi     20   0  10024   5304   3484 S   0.7   0.1   0:15.91 avahi-daemon: running [mi
```

- 35. wc :** wc stands for word count. Used for counting purpose. It is used to find out number of lines, word count, byte and characters count in the files specified in the file arguments. #wc state.txt 6 8 54 state.tx . #wc state.txt capital.txt wc -l state.txt wc -w state.txt capital.txt wc -c state.txt .wc -m state.tx

```
mits@mits-Veriton-M200-H510:~/S2$ cat Fruits
Apple
Orange
Grapes
Mango
mits@mits-Veriton-M200-H510:~/S2$ wc Fruits
 4  4 26 Fruits
```



- 36. expr :** The expr command evaluates a given expression and displays its corresponding output. It is used for: . Basic operations like addition, subtraction, multiplication, division, and modulus on integers. Evaluating regular expressions, string operations like substring, length of strings etc. Performing operations on variables inside a shell script.

```
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ grep -v 'india' india
India is my country.
All Indians are my brothers and sisters.
I love my country and I am proud of its rich and varied heritage.
I shall strive to be worthy of it.
I shall respect my parents, teachers and all elders and treat everyone with courtesy
To my country and all my people, I pledge my devotion.
In their well being and prosperity alone lies my happiness.
I am proud of INDIA
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ a=10
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ b=5
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ c=10
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ r=`expr $a + $b`
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ echo $r
15
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ r=`expr $a \* $b`
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ echo $r
50
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ r=`expr $a != $b`
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ echo $r
1
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ r=`expr $a \< $b`
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ echo $r
0
mits@mits-ThinkCentre-neo-50t-Gen-3:~$
```

- 37. Redirections & Piping :** A pipe is a form of redirection to send the output of one command/program/process to another command/program/process for further processing. Pipe is used to combine two or more commands, the output of one command acts as input to another command, and this command's output may act as input to the next command and so on.

```
mits@mits-Veriton-M200-H510:~/S2$ cat Fruits
Apple
Orange
Grapes
Mango
mits@mits-Veriton-M200-H510:~/S2$ cat Fruits | head -2 | tail -1
Orange
```

## **SHELL SCRIPT**

### **Experiment 1**

#### **AIM**

Write a shell script to read two numbers and find the sum, product, quotient and difference.

#### **PROGRAM**

```
echo "Enter two numbers"
read a
read b
sum=`expr $a + $b`
diff=`expr $a - $b`
pro=`expr $a \* $b`
quo=`expr $a / $b`
echo -e "Sum\t:" $sum
echo -e "Difference\t:" $diff
echo -e "Product\t:" $pro
echo -e "Quotient\t:" $quo
```

#### **OUTPUT**

```
$ bash operations.sh
Enter two numbers
10
5
Sum      : 15
Difference: 5
Product  : 50
Quotient : 2
```

## Experiment 2

### AIM

Write a shell script to find the largest of two numbers.

### PROGRAM

```
echo "Enter two numbers"
read a
read b
if [ $a -gt $b ]
then
echo "$a is greater then $b"
elif [ $b -gt $a ]
then
echo "$b is greater than $a"
else
echo "Both are equal"
fi
```

### OUTPUT 1

```
$ bash largein2.sh
Enter two numbers
5
6
6 is greater than 6
```

### OUTPUT 2

```
$bash largein2.sh

Enter two numbers
8
8
Both are equal
```

## Experiment 3

### AIM

Write a shell script to find the largest of three numbers.

### PROGRAM

```
echo "Enter three numbers"
read a
read b
read c
if [ $a -gt $b -a $a -gt $c ]
then
echo "$a is greater"
elif [ $b -gt $c ]
then
echo "$b is greater"
else
echo "$c is greater"
fi
```

### OUTPUT

```
$ bash largein3.sh
Enter three numbers
5
6
7
7 is greater
```

## Experiment 4

### AIM

Write a shell script to check wheather a number is divisible by 3.

### PROGRAM

```
echo "enter a number"
read a
if [ `expr $a % 3` -eq 0 ]
then
echo "$a is divisible by 3"
else
echo "$a is not divisible by 3"
fi
```

### OUTPUT 1

```
$ bash divisibleby3.sh
enter a number
6
6 is divisible by 3
```

### OUTPUT 2:

```
$ bash divisibleby3.sh
enter a number
7
7 is not divisible by 3
```

## Experiment 5

### AIM

Write a shell script to check wheather a person is eligible to vote.

### PROGRAM

```
echo "Enter the age of the person"
read age
if [ $age -ge 18 ]
then
echo "Eligible to vote"
else
echo "Not eligible to vote"
fi
```

### OUTPUT 1

```
$ bash vote.sh
Enter the age of the person
4
Not eligible to vote
```

### OUTPUT 2

```
$ bash vote.sh
Enter the age of the person
90
Eligible to vote
```

## Experiment 6


### AIM

Find the sum, average and product of the 4 integers entered

### PROGRAM

```
echo "enter 4 numbers: "
read a
read b
read c
read d
s=$((a+b+c+d))
echo sum: $s
p=`expr $a \* $b \* $c \* $d`
echo product: $p
avg=`echo "scale=2; $s / 4" | bc`
echo Average: $avg
```

### OUTPUT



```
mits@mits-Veriton-M200-H510:~/S2/pgm$ bash avg.sh
enter 4 numbers
1
2
3
4
sum: 10
product: 24
Average: 2.50
```

## Experiment 7

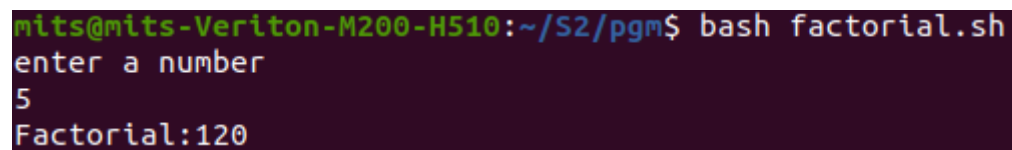
### AIM

Write a shell script to find the factorial of a given number

### PROGRAM

```
echo "Enter a number: "  
read n  
fact=1  
for((i=1;i<=n;i++))  
{  
    fact=`expr $fact \* $i`  
}  
echo "Factorial:" $fact
```

### OUTPUT



```
mits@mits-Veriton-M200-H510:~/S2/pgm$ bash factorial.sh  
enter a number  
5  
Factorial:120
```



## Experiment 8

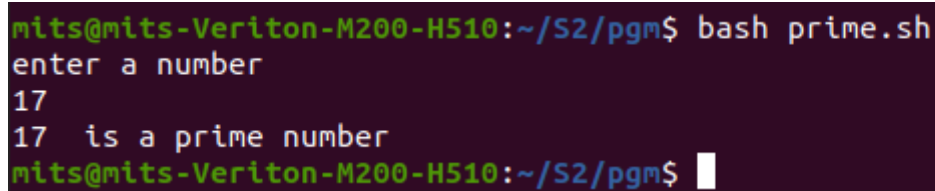
### AIM

Write a shell script to check whether a given number is prime or not

### PROGRAM

```
echo "enter a number"
read n
flag=1
for((i=2;i<n;i++))
{
x=$((n%i))
if [ $x -eq 0 ]
then
flag=0
fi
}
if [ $flag -eq 1 ]
then
echo $n " is a prime number"
else
echo $n " is not a prime number"
fi
```

### OUTPUT



```
mits@mits-Veriton-M200-H510:~/S2/pgm$ bash prime.sh
enter a number
17
17 is a prime number
mits@mits-Veriton-M200-H510:~/S2/pgm$
```

## Experiment 9

### AIM

Write a shell script to implement a simple calculator

### PROGRAM

```
echo -e "\n MENU \n ***** \n "
echo "enter 2 numbers "
read a
read b
echo -e " 1.SUM \n 2.DIFFERENE \n 3.MULTIPLICATION \n 4.DIVISION \n "
echo "enter your option"
read op
case $op in
1)res=`expr $a + $b `
echo "SUM:$res;;
2)res=`expr $a - $b `
echo "DIFFERENCE:$res;;
3)res=`expr $a \* $b `
echo "MULTIPLICATION:$res;;
4)res=`echo "scale=2; $a / $b" | bc `
echo "DIVISION:$res;;
*)echo "invalid"
esac
```

### OUTPUT

```
mits@mits-Veriton-M200-H510:~/S2/pgm$ bash calc.sh

MENU
*****

enter 2 numbers
5
2
 1.SUM
 2.DIFFERENE
 3.MULTIPLICATION
 4.DIVISION

enter your option
3
MULTIPLICATION:10
mits@mits-Veriton-M200-H510:~/S2/pgm$ bash calc.sh

MENU
*****

enter 2 numbers
6
3
 1.SUM
 2.DIFFERENE
 3.MULTIPLICATION
 4.DIVISION

enter your option
2
DIFFERENCE:3
```

## Experiment 10

### AIM

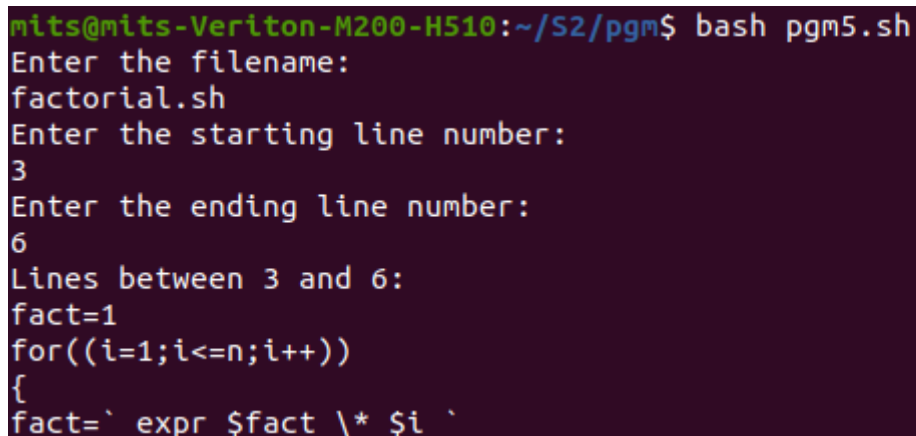
Write a shell script that accept a filename, starting and ending line numbers and display all the lines between the given line numbers

### PROGRAM

```
echo "Enter the filename:"
read f
if [ ! -f "$f" ]; then
    echo "File $f not found."
    exit 1
fi
echo "Enter the starting line number:"
read s
echo "Enter the ending line number:"
read e
if [ $s -gt $e ]; then
    echo "Error"
    exit 1
fi
l=$(wc -l < "$f")

if [ $e -gt $l ]; then
    echo "Error"
    exit 1
fi
echo "Lines between $s and $e:"
sed -n "${s},${e}p" "$f"
```

### OUTPUT



```
mits@mits-Veriton-M200-H510:~/S2/pgm$ bash pgm5.sh
Enter the filename:
factorial.sh
Enter the starting line number:
3
Enter the ending line number:
6
Lines between 3 and 6:
fact=1
for((i=1;i<=n;i++))
{
fact=`expr $fact \* $i`
```

## Experiment 11

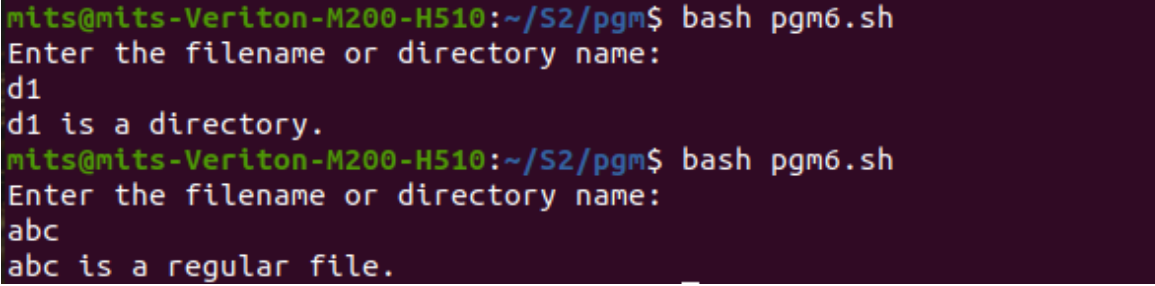
### AIM

Write a shell script that receives file names or directory names as input and check if the input provided is a file or a directory

### PROGRAM

```
echo "Enter the filename or directory name:"
read n
if [ -f "$n" ]; then
    echo "$n is a regular file."
elif [ -d "$n" ]; then
    echo "$n is a directory."
else
    echo "$n is neither a file nor a directory."
fi
```

### OUTPUT



```
mits@mits-Veriton-M200-H510:~/S2/pgm$ bash pgm6.sh
Enter the filename or directory name:
d1
d1 is a directory.
mits@mits-Veriton-M200-H510:~/S2/pgm$ bash pgm6.sh
Enter the filename or directory name:
abc
abc is a regular file.
```

## Experiment 12

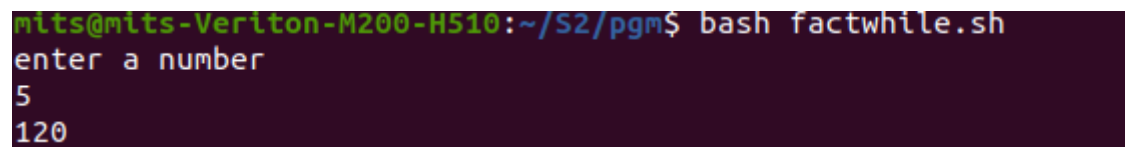
### AIM

Write a shell program to find the factorial of a given number using a while loop

### PROGRAM

```
echo "enter a number: "
read n
fact=1
i=1
while test $i -le $n
do
    fact=`expr $fact \* $i`
    i=`expr $i + 1`
done
echo $fact
```

### OUTPUT



```
mits@mits-Veriton-M200-H510:~/S2/pgm$ bash factwhile.sh
enter a number
5
120
```

## Experiment 13

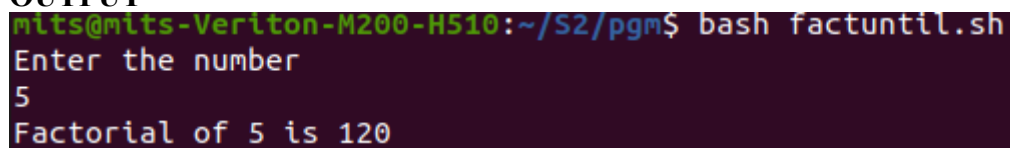
### AIM

Write a shell program to find the factorial of a given number using until loop.

### PROGRAM

```
echo "Enter the number"
read num
i=1
fact=1
until [ $i -gt $num ]
do
fact=`expr $fact \* $i`
i=`expr $i + 1`
done
echo "Factorial of $num is $fact"
```

### OUTPUT

A terminal window with a dark background. The prompt is 'mits@mits-Veriton-M200-H510:~/S2/pgm\$'. The user enters 'bash factuntil.sh'. The program outputs 'Enter the number', the user enters '5', and the program outputs 'Factorial of 5 is 120'.

```
mits@mits-Veriton-M200-H510:~/S2/pgm$ bash factuntil.sh
Enter the number
5
Factorial of 5 is 120
```

## Experiment 14

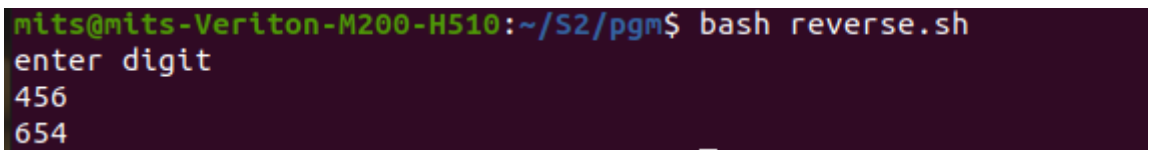
### AIM

Write a program to find reverse of a number

### PROGRAM

```
echo "enter digit"
read dig
rev=0
while [ $dig -gt 0 ]
do
i=$((dig%10))
rev=$((rev*10+i))
dig=$((dig/10))
done
echo $rev
```

### OUTPUT



```
mits@mits-Veriton-M200-H510:~/S2/pgm$ bash reverse.sh
enter digit
456
654
```

## Experiment 15

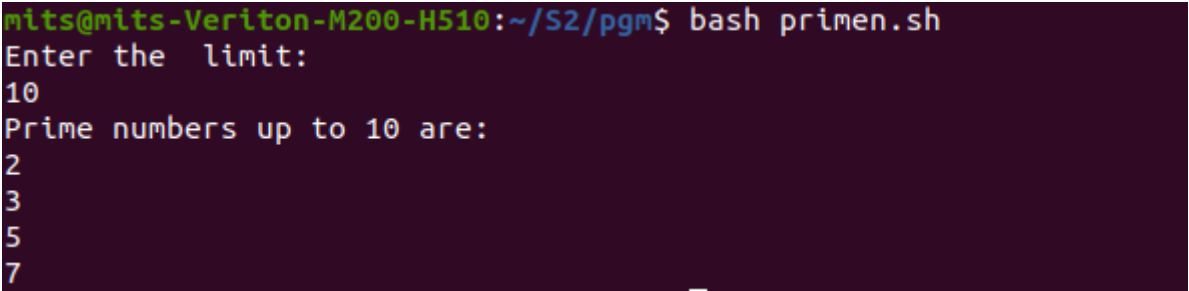
### AIM

Write a shell script to display prime numbers within a specific limit

### PROGRAM

```
echo "Enter the limit:"
read l
if [ $l -lt 2 ]; then
    echo "Error."
    exit 0
fi
echo "Prime numbers up to $l are:"
for (( num=2; num<=l; num++ )); do
    is_prime=true
    for (( i=2; i*i<=num; i++ )); do
        if [ $((num%i)) -eq 0 ]; then
            is_prime=false
            break
        fi
    done
    if [ $is_prime = true ]; then
        echo $num
    fi
done
```

### OUTPUT



```
mits@mits-Veriton-M200-H510:~/S2/pgm$ bash primen.sh
Enter the limit:
10
Prime numbers up to 10 are:
2
3
5
7
```



## Experiment 16

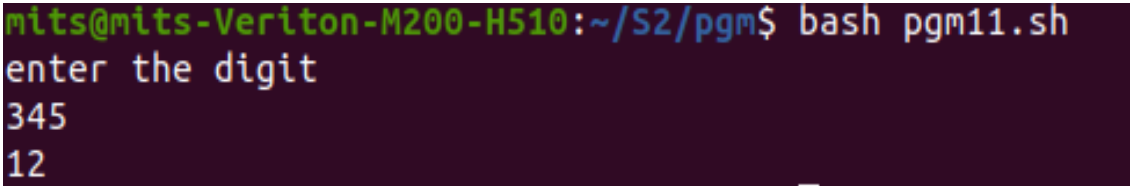
### AIM

Write a shell script to find the sum of digits of a number using any loop

### PROGRAM

```
echo "enter the digit"
read a
i=1
n=0
while [ $a -gt 0 ]
do
n=`expr $a % 10 `
a=`expr $a / 10 `
z=`expr $z + $n `
done
echo "$z"
```

### OUTPUT



```
mits@mits-Veriton-M200-H510:~/S2/pgm$ bash pgm11.sh
enter the digit
345
12
```

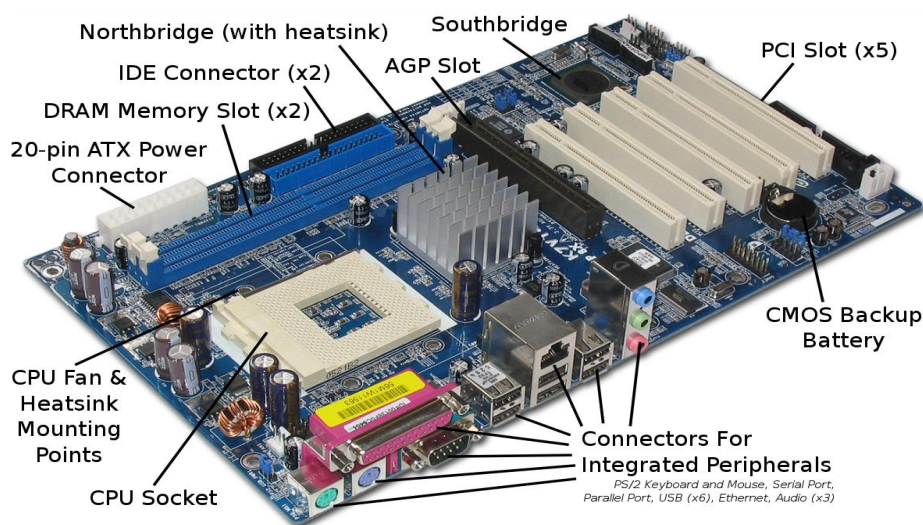
# INTRODUCTION TO COMPUTER HARDWARES

Physical identification of major components of a computer system such as motherboard, RAM modules, daughter cards, bus slots, SMPS, internal storage devices, interfacing ports. Specifications of desktop and server class computers. Installation of common operating systems for desktop and server use. (Students may be asked to formulate specification for computer to be used as Desktop, Web server)

## MOTHERBOARD

A motherboard (also called mainboard, main circuit board, or mobo) is the main printed circuit board (PCB) in general-purpose computers and other expandable systems. It holds and allows communication between many of the crucial electronic components of a system, such as the central processing unit (CPU) and memory, and provides connectors for other peripherals. Unlike a backplane, a motherboard usually contains significant sub-systems, such as the central processor, the chipset's input/output and memory controllers, interface connectors, and other components integrated for general use.

Motherboard means specifically a PCB with expansion capabilities. As the name suggests, this board is often referred to as the "mother" of all components attached to it, which often include peripherals, interface cards, and daughterboards: sound cards, video cards, network cards, host bus adapters, TV tuner cards, IEEE 1394 cards; and a variety of other custom components.



## RAM MODULES

In computing, a memory module or RAM (random-access memory) stick is a printed circuit board on which memory integrated circuits are mounted. Memory modules permit easy installation and replacement in electronic systems, especially computers such as personal computers, workstations, and servers. The first memory modules were proprietary designs that were specific to a model of computer from a specific manufacturer. Later, memory modules were standardized by organizations such as JEDEC and could be used in any system designed to use them.

Types of memory module include:

- TransFlash Memory Module
- SIMM, a single in-line memory module
- DIMM, dual in-line memory module
- Rambus memory modules are a subset of DIMMs, but are normally referred to as RIMMs
- SO-DIMM, small outline DIMM, a smaller version of the DIMM, used in laptops

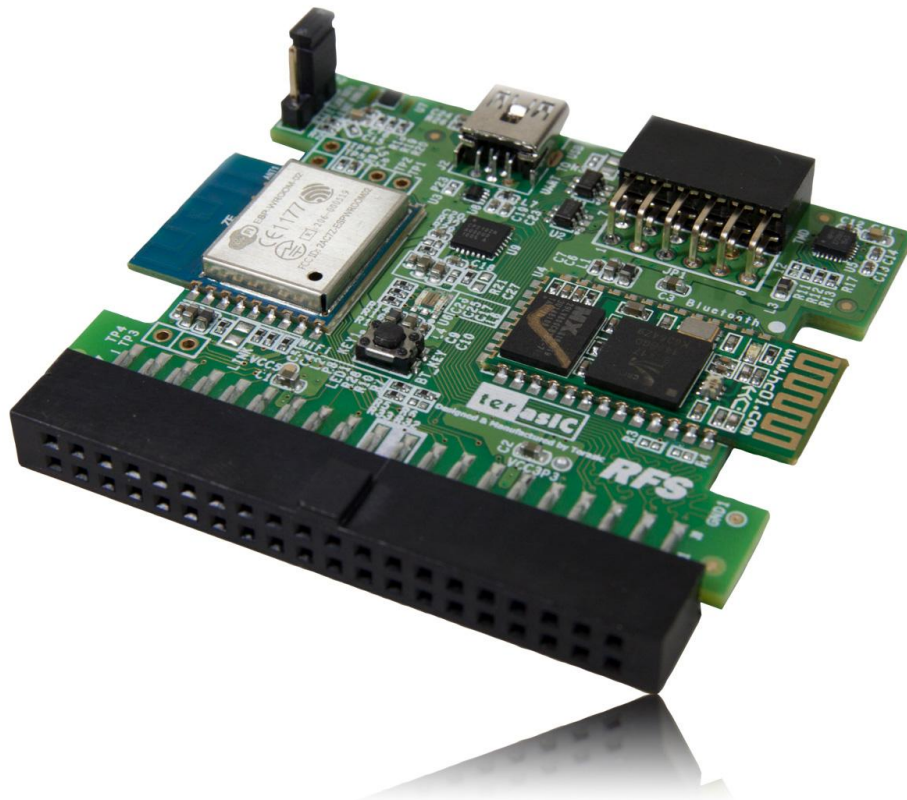
Distinguishing characteristics of computer memory modules include voltage, capacity, speed (i.e, bit rate), and form factor. For economic reasons, the large (main) memories found in personal computers, workstations, and non-handheld game-consoles (such as PlayStation and Xbox) normally consist of dynamic RAM (DRAM). Other parts of the computer, such as cache memories normally use static RAM (SRAM). Small amounts of SRAM are sometimes used in the same package as DRAM. [2] However, since SRAM has high leakage power and low density, die-stacked DRAM has recently been used for designing multi-megabyte sized processor caches.



## DAUGHTER BOARD

The daughter board is a computer hardware. It is also known as the piggyback board, riser card, daughter board, daughtercard or daughter card. A daughter board is a printed circuit board which is connected to the motherboard or expansion card. As compared to the motherboard, it is smaller in size. A daughter board does not act as an expansion card. An expansion card adds extra new functions to the computer. But a daughter board that is connected to the motherboard adds or supports the main functions of the motherboard.

Daughter boards are directly connected to the motherboards. You know that expansion cards are connected to the motherboard by using the bus and other serial interfaces. But the daughter board is directly connected to the board by soldering. As an update of the motherboard or expansion card, daughter boards are released to extend the features and services of the motherboard or expansion cards.



## BUS SLOTS

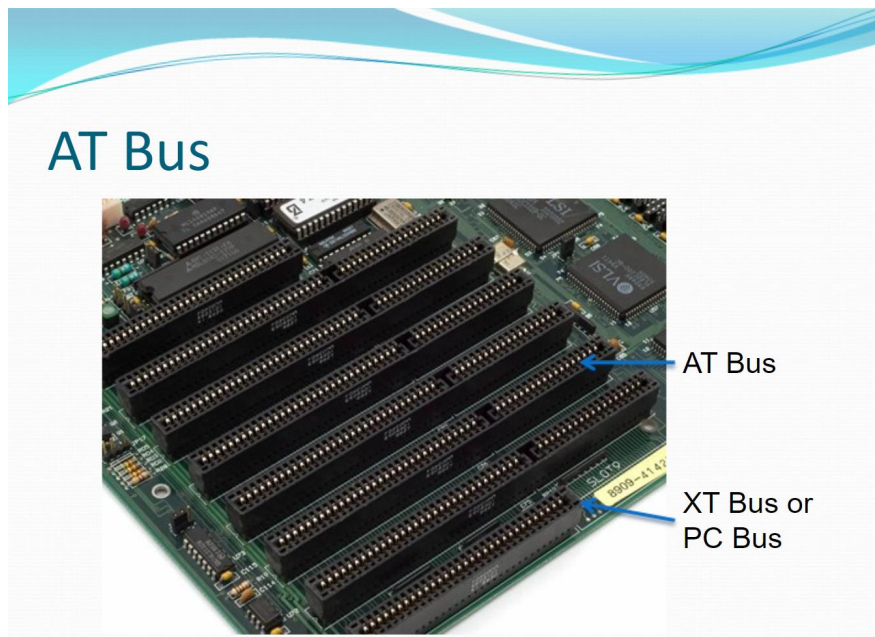
Alternatively known as a bus slot or expansion port, an expansion slot is a connection or port inside a computer on the motherboard or riser card. It provides an installation point

for a hardware expansion card to be connected, which provides additional features to a computer such as video, sound, advanced graphics, Ethernet or memory.

The expansion card has an edge connector that fits precisely into the expansion slot as well as a row of contacts that is designed to establish an electrical connection between the motherboard and the electronics on the card, which are mostly integrated circuits. Depending on the formfactor of the case and motherboard, a computer system generally can have anywhere from one to seven expansion slots. With a backplane system, up to 19 expansion cards can be installed.

Expansion cards can provide various functions including

- Sound
- Modems
- Network
- Interface adapters
- TV and radio tuning
- Video processing
- Host adapting such as redundant array of independent disks or
- small computer system interface
- Solid-state drive
- Power-on self-test
- Advanced multi rate codec
- Basic input/output system
- (BIOS)
- Expansion read-only memory
- (ROM)
- Security devices
- RAM memory





## SMPS

A switched-mode power supply (SMPS) is an electronic circuit that converts power using switching devices that are turned on and off at high frequencies, and storage components such as inductors or capacitors to supply power when the switching device is in its non- conduction state.

Switching power supplies have high efficiency and are widely used in a variety of electronic equipment, including computers and other sensitive equipment requiring stable and efficient power supply.

A switched-mode power supply is also known as a switch-mode power supply or switching-mode power supply.

Switched-mode power supplies are classified according to the type of input and output voltages.

The four major categories are

- AC to DC
- DC to DC
- DC to AC
- AC to AC

A basic isolated AC to DC switched-mode power supply consists of:

- Input rectifier and filter
- Inverter consisting of switching devices such as MOSFETS
- Transformer
- Output rectifier and filter
- Feedback and control circuit



## INTERNAL STORAGE DEVICES

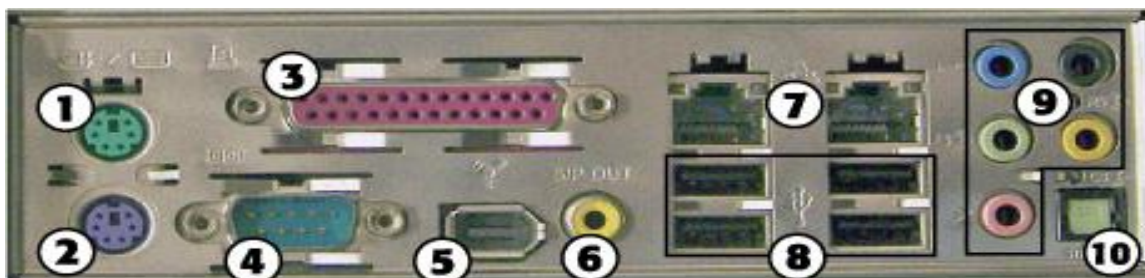


A storage unit is a part of the computer system which is employed to store the information and instructions to be processed. A storage device is an integral part of the computer hardware which stores information/data to process the result of any computational work. Without a storage device, a computer would not be able to run or even boot up. Or in other words, we can say that a storage device is hardware that is used for storing, porting, or extracting data files. It can also store information/data both temporarily and permanently. Computer storage is of two types:

**Primary Storage Devices:** It is also known as internal memory and main memory. This is a section of the CPU that holds program instructions, input data, and intermediate results. It is generally smaller in size. RAM (Random Access Memory) and ROM (Read Only Memory) are examples of primary storage.

**Secondary Storage Devices:** Secondary storage is a memory that is stored external to the computer. It is mainly used for the permanent and long-term storage of programs and data. Hard Disk, CD, DVD, Pen/Flash drive, SSD, etc., are examples of secondary storage.

## INTERFACING PORTS



- |                       |                                      |
|-----------------------|--------------------------------------|
| 1. PS/2 mouse port    | 6. SPDIF coaxial digital audio port  |
| 2. PS/2 keyboard port | 7. Ethernet ports                    |
| 3. Parallel port      | 8. USB ports                         |
| 4. Serial port        | 9. 1/8-inch mini-jack audio ports    |
| 5. IEEE 1394a port    | 10. SPDIF optical digital audio port |

A port is basically a physical docking point which is basically used to connect the external devices to the computer, or we can say that A port acts as an interface between the computer and the external devices, e.g., we can connect hard drives, printers to the computer with the help of ports.

### Characteristics of Ports

- A port has the following characteristics -
- External devices are connected to a computer using cables and ports.
- Ports are slots on the motherboard into which a cable of an external device is plugged in.
- Examples of external devices attached via ports are the mouse, keyboard, monitor, microphone, speakers, etc.



# LAMP INSTALLATION PROCEDURE

## 1. Install Apache

- Updating the local package index to reflect the latest upstream changes:

**sudo apt update && sudo apt install ufw**

```
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ sudo apt update && sudo apt install ufw
[sudo] password for mits:
Get:1 https://packages.microsoft.com/repos/code stable InRelease [3,590 B]
Get:2 https://packages.microsoft.com/repos/code stable/main arm64 Packages [17.0 kB]
Get:3 https://packages.microsoft.com/repos/code stable/main armhf Packages [16.8 kB]
Get:4 https://packages.microsoft.com/repos/code stable/main amd64 Packages [16.8 kB]
Hit:5 http://in.archive.ubuntu.com/ubuntu jammy InRelease
Get:6 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:7 http://in.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Hit:8 http://in.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:9 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1,343 kB]
Get:10 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1,562 kB]
Get:11 http://security.ubuntu.com/ubuntu jammy-security/main i386 Packages [444 kB]
Get:12 http://in.archive.ubuntu.com/ubuntu jammy-updates/main i386 Packages [610 kB]
Get:13 http://in.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [297 kB]
Get:14 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [237 kB]
Get:15 http://in.archive.ubuntu.com/ubuntu jammy-updates/restricted i386 Packages [36.5 kB]
Get:16 http://security.ubuntu.com/ubuntu jammy-security/restricted i386 Packages [33.6 kB]
Get:17 http://in.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1,706 kB]
Get:18 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1,662 kB]
Get:19 http://in.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [286 kB]
Get:20 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [280 kB]
Get:21 http://in.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1,060 kB]
Get:22 http://security.ubuntu.com/ubuntu jammy-security/universe i386 Packages [599 kB]
Get:23 http://in.archive.ubuntu.com/ubuntu jammy-updates/universe i386 Packages [698 kB]
Get:24 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [852 kB]
Get:25 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [37.2 kB]
Get:26 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [7,588 B]
Get:27 http://in.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [49.7 kB]
Get:28 http://in.archive.ubuntu.com/ubuntu jammy-updates/multiverse Translation-en [12.1 kB]
Fetched 12.1 MB in 8s (1,462 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
292 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ufw is already the newest version (0.36.1-4ubuntu0.1).
0 upgraded, 0 newly installed, 0 to remove and 292 not upgraded.
mits@mits-ThinkCentre-neo-50t-Gen-3:~$
```

- Then, install the apache2 package:

**sudo apt install apache2**

```
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
apache2 is already the newest version (2.4.52-1ubuntu4.8).
0 upgraded, 0 newly installed, 0 to remove and 292 not upgraded.
mits@mits-ThinkCentre-neo-50t-Gen-3:~$
```

- Install Apache using apt: It is recommended that you enable the most restrictive profile that will still allow the traffic You have configured. Since we haven't configured SSL for our server yet in this guide, we will only need to allow traffic on port 80:

**sudo ufw allow 'Apache'**

```
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ sudo ufw allow 'Apache'
Skipping adding existing rule
Skipping adding existing rule (v6)
mits@mits-ThinkCentre-neo-50t-Gen-3:~$
```

- You can verify the change by typing: **sudo ufw status**. The output will provide a list of allowed HTTP traffic.

```
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ sudo ufw status
Status: active

To Action From
--
Apache ALLOW Anywhere
80 ALLOW Anywhere
Apache (v6) ALLOW Anywhere (v6)
80 (v6) ALLOW Anywhere (v6)

mits@mits-ThinkCentre-neo-50t-Gen-3:~$
```

### Checking your Web Server

- At the end of the installation process, Ubuntu 20.04 starts Apache. The web server should already be up and running. Check with the systemd init system to make sure the service is running by typing:

**sudo systemctl status apache2**

```
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor prese
   Active: active (running) since Thu 2024-04-11 13:32:12 IST; 5h 16min left
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 916 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUC
 Main PID: 986 (apache2)
    Tasks: 55 (limit: 9038)
   Memory: 7.8M
      CPU: 61ms
   CGroup: /system.slice/apache2.service
           └─986 /usr/sbin/apache2 -k start
             └─988 /usr/sbin/apache2 -k start
               └─989 /usr/sbin/apache2 -k start

Apr 11 13:32:12 mits-ThinkCentre-neo-50t-Gen-3 systemd[1]: Starting The Apache
Apr 11 13:32:12 mits-ThinkCentre-neo-50t-Gen-3 apachectl[942]: AH00558: apache2
Apr 11 13:32:12 mits-ThinkCentre-neo-50t-Gen-3 systemd[1]: Started The Apache H
lines 1-17/17 (END)
```

- We can access the default Apache landing page to confirm that the software is running properly through your IP address. If you do not know your server's IP address, you can get it a few different ways from the command line.
- Try typing this at your server's command prompt:

**hostname -I.**



```

mits@mits-ThinkCentre-neo-50t-Gen-3:~$ hostname -l
hostname: invalid option -- 'l'
Usage: hostname [-b] {hostname|-F file}          set host name (from file)
           hostname [-a|-A|-d|-f|-i|-I|-s|-y]      display formatted name
           hostname                                     display host name

           {yp,nis,}domainname {nisdomain|-F file} set NIS domain name (from file)
           {yp,nis,}domainname                     display NIS domain name

           dnsdomainname                             display dns domain name

           hostname -V|--version|-h|--help         print info and exit

Program name:
           {yp,nis,}domainname=hostname -y
           dnsdomainname=hostname -d

Program options:
  -a, --alias                alias names
  -A, --all-fqdns            all long host names (FQDNs)
  -b, --boot                 set default hostname if none available
  -d, --domain               DNS domain name
  -f, --fqdn, --long         long host name (FQDN)
  -F, --file                 read host name or NIS domain name from given file
  -i, --ip-address           addresses for the host name
  -I, --all-ip-addresses     all addresses for the host
  -s, --short                short host name
  -y, --yp, --nis            NIS/YP domain name

Description:
  This command can get or set the host name or the NIS domain name. You can
  also get the DNS domain or the FQDN (fully qualified domain name).
  Unless you are using bind or NIS for host lookups you can change the
  FQDN (Fully Qualified Domain Name) and the DNS domain name (which is
  part of the FQDN) in the /etc/hosts file.
mits@mits-ThinkCentre-neo-50t-Gen-3:~$

```

- We will get back a few addresses separated by spaces. You can try each in your web browser to determine if they work.
- Another option is to use the `icanhazip` tool, which should give you your public IP address as read from another location on the internet:

**`curl -4 icanhazip.com`**

```

mits@mits-ThinkCentre-neo-50t-Gen-3:~$ curl -4 icanhazip.com
103.214.233.56
mits@mits-ThinkCentre-neo-50t-Gen-3:~$

```

- When you have your server's IP address, enter it into your browser's address bar: [http://your\\_server\\_ip](http://your_server_ip) and you should see the default Ubuntu 20.04 Apache web page.

## 2. Install mariadb

- Install mariaDB

**sudo apt install mariadb-server mariadb-client**

```
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ sudo apt install mariadb-server mariadb-client
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
mariadb-client is already the newest version (1:10.6.16-0ubuntu0.22.04.1).
mariadb-server is already the newest version (1:10.6.16-0ubuntu0.22.04.1).
0 upgraded, 0 newly installed, 0 to remove and 292 not upgraded.
mits@mits-ThinkCentre-neo-50t-Gen-3:~$
```

- Check mariadb Installation
- `sudo systemctl status mysql` (if it is not working `sudo systemctl start mysql` )

## 3. Install PHP

- Install phpmyadmin

**sudo apt install phpmyadmin**

```
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ sudo apt install phpmyadmin
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
phpmyadmin is already the newest version (4:5.1.1+dfsg1-5ubuntu1).
0 upgraded, 0 newly installed, 0 to remove and 292 not upgraded.
mits@mits-ThinkCentre-neo-50t-Gen-3:~$
```

- ( It will ask for web server select apache2, select dbconfiguration and set password )
- Restart apache2

**sudo systemctl restart apache2**

```
mits@mits-ThinkCentre-neo-50t-Gen-3:~$ sudo systemctl restart apache2
mits@mits-ThinkCentre-neo-50t-Gen-3:~$
```

# NETWORK COMMANDS

Network commands in Windows - ping, route, traceroute, nslookup, IpConfig, NetStat.

- **Ping**

Ping is an old Unix tool that has been around for a long time but many PC users are unfamiliar with the Windows version. Ping sends out a packet to a designated internet host or network computer and measures its response time.

```
C:\Users\mutho>ping google.com

Pinging google.com [142.250.195.142] with 32 bytes of data:
Reply from 142.250.195.142: bytes=32 time=14ms TTL=117
Reply from 142.250.195.142: bytes=32 time=16ms TTL=117
Reply from 142.250.195.142: bytes=32 time=24ms TTL=117
Reply from 142.250.195.142: bytes=32 time=19ms TTL=117

Ping statistics for 142.250.195.142:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 14ms, Maximum = 24ms, Average = 18ms

C:\Users\mutho>ping geeksforgeeks.com

Pinging geeksforgeeks.com [199.59.243.225] with 32 bytes of data:
Reply from 199.59.243.225: bytes=32 time=18ms TTL=248
Reply from 199.59.243.225: bytes=32 time=16ms TTL=248
Reply from 199.59.243.225: bytes=32 time=22ms TTL=248
Reply from 199.59.243.225: bytes=32 time=31ms TTL=248

Ping statistics for 199.59.243.225:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 16ms, Maximum = 31ms, Average = 21ms
```

- **Route**

In computing, route is a command used to view and manipulate the IP routing table in Unix-like and Microsoft Windows[1] operating systems and also in



IBM OS/2 and ReactOS.[2] Manual manipulation of the routing table is characteristic of static routing.

```
C:\Users\mutho>route print
=====
Interface List
 14...e0 be 03 93 d2 1f .....Intel(R) Ethernet Connection (17) I219-V
 1.....Software Loopback Interface 1
=====

IPv4 Route Table
=====
Active Routes:
Network Destination        Netmask          Gateway          Interface        Metric
0.0.0.0                    0.0.0.0          10.76.0.1        10.76.8.13       281
10.76.0.0                  255.255.224.0    On-link          10.76.8.13       281
10.76.8.13                  255.255.255.255  On-link          10.76.8.13       281
10.76.31.255                255.255.255.255  On-link          10.76.8.13       281
127.0.0.0                  255.0.0.0        On-link          127.0.0.1        331
127.0.0.1                  255.255.255.255  On-link          127.0.0.1        331
127.255.255.255            255.255.255.255  On-link          127.0.0.1        331
224.0.0.0                  240.0.0.0        On-link          127.0.0.1        331
224.0.0.0                  240.0.0.0        On-link          10.76.8.13       281
255.255.255.255            255.255.255.255  On-link          127.0.0.1        331
255.255.255.255            255.255.255.255  On-link          10.76.8.13       281
=====
Persistent Routes:
Network Address            Netmask  Gateway Address  Metric
0.0.0.0                    0.0.0.0    10.76.0.1      Default
=====
```

- **Nslookup**

This command helps diagnose the Domain Name System (DNS) infrastructure and comes with a number of sub-commands. These are mainly for systems administrators. The primary interest for average PC users is its use to find the computer name corresponding to a numeric IP. For example, if you want to know who is "216.109.112.135" , enter "nslookup 216.109.112.135" and you will find that it is (or was anyway) a Yahoo computer. My firewall keeps a log of the IPs involved in the attempts to probe my computer and I sometimes look a few up to see who they are. (There are also Whois search sites available on the Web as mentioned in the Ipconfig section.)

```
C:\Users\mutho>nslookup google.com
Server:  dns.google
Address:  8.8.8.8

Non-authoritative answer:
Name:     google.com
Addresses: 2404:6800:4007:81b::200e
          142.250.195.142

C:\Users\mutho>nslookup 209.132.183.181
Server:  dns.google
Address:  8.8.8.8

*** dns.google can't find 209.132.183.181: Non-existent domain
C:\Users\mutho>
```

- **Ipconfig**

The Windows IP Configuration tool (ipconfig) is the command-line equivalent of the accessory "Winipcfg" that was present in Windows 9X/Me. It is used to display the TCP/IP network configuration values. To open it, enter "ipconfig" in the command prompt. If you are connected directly to the Internet, you will obtain your IP address.

```
C:\Users\mutho>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : fe80::9c26:3b6c:4e46:cb60%14
    IPv4 Address. . . . . : 10.76.8.13
    Subnet Mask . . . . . : 255.255.224.0
    Default Gateway . . . . . : 10.76.0.1

C:\Users\mutho>
```



- **Tracert**

Tracert (traceroute) is another old tool borrowed from Unix. The actual path between two computers on the Internet is not a straight line but consists of numerous segments or "hops" from one intermediate computer to another. Tracert shows each step of the path taken. It can be interesting to see just how convoluted it is. The times for each hop and the IP addresses for each intermediate computer are displayed. Tracert shows up to 30 hops. It is convenient for finding if there is one particular segment that is causing a slow or bad connection. A typical command might be "tracert dell.com".

```
C:\Users\mutho>tracert google.com

Tracing route to google.com [142.250.196.14]
over a maximum of 30 hops:

  0  <1 ms    <1 ms    <1 ms    10.76.0.1
  1  4 ms     4 ms     7 ms     103.214.233.1
  2  2 ms     6 ms     5 ms     103.214.235.243
  3  14 ms    13 ms    14 ms    undefined.hostname.localhost [103.179.111.177]
  4  17 ms    18 ms    14 ms    10.200.150.2
  5  16 ms    16 ms    16 ms    72.14.222.8
  6  16 ms    18 ms    16 ms    216.239.43.133
  7  19 ms    18 ms    15 ms    142.251.55.43
  8  15 ms    14 ms    14 ms    maa03s44-in-f14.1e100.net [142.250.196.14]

Trace complete.
```

- **Identify and perform 5 more network commands and it's working.**

- a. arp**

The ARP command corresponds to the Address Resolution Protocol. Although it is easy to think of network communications in terms of IP addressing, packet delivery is ultimately dependent on the Media Access

Control (MAC) address of the device's network adapter. This is where the Address Resolution Protocol comes into play. Its job is to map IP addresses to MAC addresses. Windows devices maintain an ARP cache, which contains the results of recent ARP queries. You can see the contents of this cache by using the ARP -A command. If you are having problems communicating with one specific host, you can append the remote host's IP address to the ARP -A command.

```
C:\Users\mutho>arp -g

Interface: 10.76.8.13 --- 0xe
    Internet Address      Physical Address      Type
    10.76.0.1             70-4c-a5-36-de-7e    dynamic
    10.76.31.255          ff-ff-ff-ff-ff-ff    static
    224.0.0.22            01-00-5e-00-00-16    static
    224.0.0.251          01-00-5e-00-00-fb    static
    224.0.0.252          01-00-5e-00-00-fc    static
    239.255.255.250      01-00-5e-7f-ff-fa    static

C:\Users\mutho>arp -a

Interface: 10.76.8.13 --- 0xe
    Internet Address      Physical Address      Type
    10.76.0.1             70-4c-a5-36-de-7e    dynamic
    10.76.31.255          ff-ff-ff-ff-ff-ff    static
    224.0.0.22            01-00-5e-00-00-16    static
    224.0.0.251          01-00-5e-00-00-fb    static
    224.0.0.252          01-00-5e-00-00-fc    static
    239.255.255.250      01-00-5e-7f-ff-fa    static
```

## b. nbtstat

The netstat command is like a special tool in Linux that helps you understand and check things about how your computer connects to the internet. It can tell you about the connections your computer is making, the paths it uses to send information, and even some technical details like how many packets of data are being sent or received. In simple terms, it's like a window that shows you what's happening with your computer and the internet. This article will help you learn how to use netstat, exploring different ways to get specific information and giving you a better idea of what's going on behind the scenes.

```
C:\Users\mutho>nbtstat -r

NetBIOS Names Resolution and Registration Statistics
-----

Resolved By Broadcast      = 0
Resolved By Name Server    = 0

Registered By Broadcast    = 24
Registered By Name Server  = 0

C:\Users\mutho>
```

### c.hostname

The previously discussed NbtStat command can provide you with the host name that has been assigned to a Windows device, if you know which switch to use with the command. However, if you're just looking for a fast and easy way of verifying a computer's name, then try using the Hostname command. Typing Hostname at the command prompt returns the local computer name.

```
C:\Users\mutho>hostname
DESKTOP-ENPI76M
```

### d. pathping

Pathping command is a combination of ping and tracert command. It sends request to each routers that comes between source and destination and compute result based on response from each router. It provide continues monitoring of the network path which allow network administrator to observe changes in performance.

```
C:\Users\mutho>pathping 172.16.0.254

Tracing route to 172.16.0.254 over a maximum of 30 hops

  0  DESKTOP-ENPI76M [10.76.8.13]
  1  10.76.0.1
  2  103.214.233.1
  3  103.214.235.243
  4  * * *
Computing statistics for 75 seconds...
Hop  RTT      Source to Here   This Node/Link   Address
  0                               Lost/Sent = Pct  Lost/Sent = Pct
  0                               0/ 100 = 0%      0/ 100 = 0%      DESKTOP-ENPI76M [10.76.8.13]
  1    0ms      0/ 100 = 0%      0/ 100 = 0%      | 10.76.0.1
  2    4ms      0/ 100 = 0%      0/ 100 = 0%      | 103.214.233.1
  3    3ms      0/ 100 = 0%      0/ 100 = 0%      | 103.214.235.243
Trace complete.
```

## e.getmac

Another very simple command that shows the MAC address of your network interfaces.

```
C:\Users\mutho>getmac

Physical Address    Transport Name
=====
E0-BE-03-93-D2-1F  \Device\NPF{FE97DF42-1509-4D6E-A103-5DA53F45B000}
C:\Users\mutho>
```