S.B. JAIN INSTITUTE OF TECHNOLOGY MANAGEMENT & RESEARCH, NAGPUR

Practical 02

Aim: To understand and demonstrate the use of basic commands in different operating systems (Windows, Linux, and UNIX) for managing files, directories, permissions, and user interactions through a terminal or command-line interface.

Name: Sayali Shamrao Misal

USN: CM25D007

Semester/Year: IV/II

Academic Session: 2025-2026

Date of Performance: 20/01/2026

Date of Submission: /01/2026

❖ **Aim:** To understand and demonstrate the use of basic commands in different operating systems (Windows, Linux, and UNIX) for managing files, directories, permissions, and user interactions through a terminal or command-line interface.

❖ Objectives:

1. To learn and practice fundamental command-line operations for file and directory management.
2. To explore and utilize user and permission management commands effectively.
3. To enhance system administration skills by working with commands across different operating systems.

❖ Requirements:

✓ Hardware Requirements:

- **Processor**: Multi-core CPU, Intel Core i3 (3.0 GHz) or higher
- **RAM**: Minimum 4 GB (8 GB recommended for optimal performance)
- **Storage**: 100 GB HDD or SSD (Solid State Drive) for faster access
- **Network Interface**: Ethernet or Wi-Fi adapter for connectivity

✓ Software Requirements:

- **Operating System**: Windows 10/11, Linux (Ubuntu 20.04/CentOS 8), UNIX-based OS
- **Command-line Interface**: PowerShell or Command Prompt (Windows), Terminal (Linux/UNIX)
- **Text Editor**: Nano, Vim, or Visual Studio Code for file editing
- **Administrative Privileges**: Superuser (Linux/UNIX) or Administrator (Windows) access

❖ Theory:

In system administration, command-line interfaces (CLI) are essential tools for managing and interacting with operating systems like Windows, Linux, and UNIX. Commands allow users to perform various tasks such as navigating directories, managing files, controlling permissions, and monitoring system performance. Each operating system provides a set of built-in commands, such as ‘man’, ‘ls’, ‘cd’, ‘mkdir’, and ‘chmod’, to facilitate efficient system management. Understanding these commands and their syntax is crucial for automating tasks, enhancing security, and ensuring optimal system functionality. This practical aims to develop foundational skills in executing and applying basic commands across different platforms.

❖ Commands:

1) **Display User Manual of a Command**
   - Functionality: Shows the manual page with details about a command’s usage, options, and arguments.
   - Syntax: man <command>
   - Example: man ls

2) **Change Current Working Directory**.
   - Functionality: Changes the terminal’s current working directory.
   - Syntax: cd <directory-path>
   - Example: cd /home/user/Documents.

3) **List Contents of the Current Directory.**
   - Functionality: Lists all files and directories in the current location.
   - Syntax: ls
   - Example: ls

4) **Read/Modify/Concatenate Text Files.**
   - Functionality: Displays or manipulates file content.
   - Syntax:
     - Read: cat <filename>
     - Modify: ‘nano <filename>
     - Concatenate: cat <file1> <file2> > <outputfile>

   Create a New Directory.
   - Functionality: Creates a new directory at the specified path.
   - Syntax: mkdir <directory-name
   - Example: mkdir newdir

5) **Display Current Working Directory.**
   - Functionality: Prints the current directory path.
   - Syntax: pwd
   - Example: pwd

6) **Write Arguments to Standard Output.**
   - Functionality: Prints the provided string or variables.
   - Syntax: echo <arguments>
   - Example: echo Hello World

7) **Remove a File**.mkd
   - Functionality: Deletes a specified file.
   - Syntax: rm <filename>
   - Example: rm file.txt

8) **Delete a Directory.**
   - Functionality: Removes an empty directory.
   - Syntax: rmdir <directory-name>
   - Example: rmdir olddir

9) **Copy a File or Directory.**
   - Functionality: Copies a file or directory to a destination.
   - Syntax: cp <source> <destination>
   - Example: cp file.txt backup/

10) **Switch to Root User.**
   - Functionality: Gains root privileges temporarily.
   - Syntax: sudo su
   - Example: sudo s

11) **Move Files or Directories.**
   - Functionality: Moves or renames files and directories.
   -Syntax: mv <sosudource> <destination>
   - Example: mv file.txt newdir/

12) **Search for a String in a File.**
   - Functionality: Searches for a specific word or pattern in a file.
   - Syntax: grep "<string>" <file>
   - Example: grep "error" log.txt

13) **Print Top N Lines of a File.**
   - Functionality: Displays the first N lines of a file.
   - Syntax: head -n <N> <file>
   - Example: ‘head -n 10 file.txt’

14) **Print Last N Lines of a File.**
   - Functionality: Displays the last N lines of a file.
   - Syntax: tail -n <N> <file>
   - Example: ‘tail -n 10 file.txt

15) **Remove Read Permission from Owner.**
   - Functionality: Revokes the owner’s read permission for a file.
   - Syntax: chmod u-r <filename>
   - Example: chmod u-r file.txt

16) **Change Specific Permissions.**
   - Functionality: Sets or removes specific file permissions.
   -Syntax: chmod u+r,w-x,g+w <filename>
   - Example: chmod u+r,w-x,g+w file.txt

17) **Add Write Permission to Owner, None to Others**.
   -Functionality: Allows write access for the owner only.
   - Syntax: chmod u+w,o-rwx <filename>
   - Example: chmod u+w,o-rwx file.txt

18) **Assign Permissions to Users.**
   - Functionality: Modifies file access for users, groups, and others.
   - Syntax: chmod u+rwx,g+rx,o+r <filename>
   - Example: ‘chmod u+rwx,g+rx,o+r file.txt

19) **Assign R/W/X to Others.**
   - Functionality: Gives read, write, and execute permissions to others.
   - Syntax: chmod o+rwx <filename>
   - Example: chmod o+rwx file.txt

20) **Remove All Permissions from All Users.**
   - Functionality: Clears all permissions on a file.
   - Syntax: ‘chmod a-rwx <filename>
   - Example: ‘chmod a-rwx file.txt

21) **Remove Read Permission Using Absolute Mode.**
   - Functionality: Uses numeric mode to restrict read access.
   - Syntax: chmod 700 <filename>
   - Example: chmod 700 file.txt

22) **Set R/W for Owner, None for Group/Other.**
   - Functionality: Assigns permissions in numeric mode.
   - Syntax: chmod 600 <filename>
   - Example: chmod 600 file.txt’

23) **Add Execute for Owner, Read for Group/Others.**
   - Functionality: Adds execution and read access.
   - Syntax: chmod u+x,g+r,o+r <filename>

- Example: chmod u+x,g+r,o+r file.txt

24) **Add Execute Permission to All Users.**
- Functionality: Enables execution by everyone.
- Syntax: chmod a+x <filename>
- Example: chmod a+x script.sh

❖ **Conclusion:** In conclusion, understanding and using essential operating system commands like **'ls', 'cd', 'cp', 'mv', and 'chmod'** enables efficient file management, navigation, and permission control. Tools like **'grep', 'head', and 'tail'** enhance data processing. Mastery of these commands improves system administration, task automation, and overall system security and performance.

❖ **Discussion Questions:**

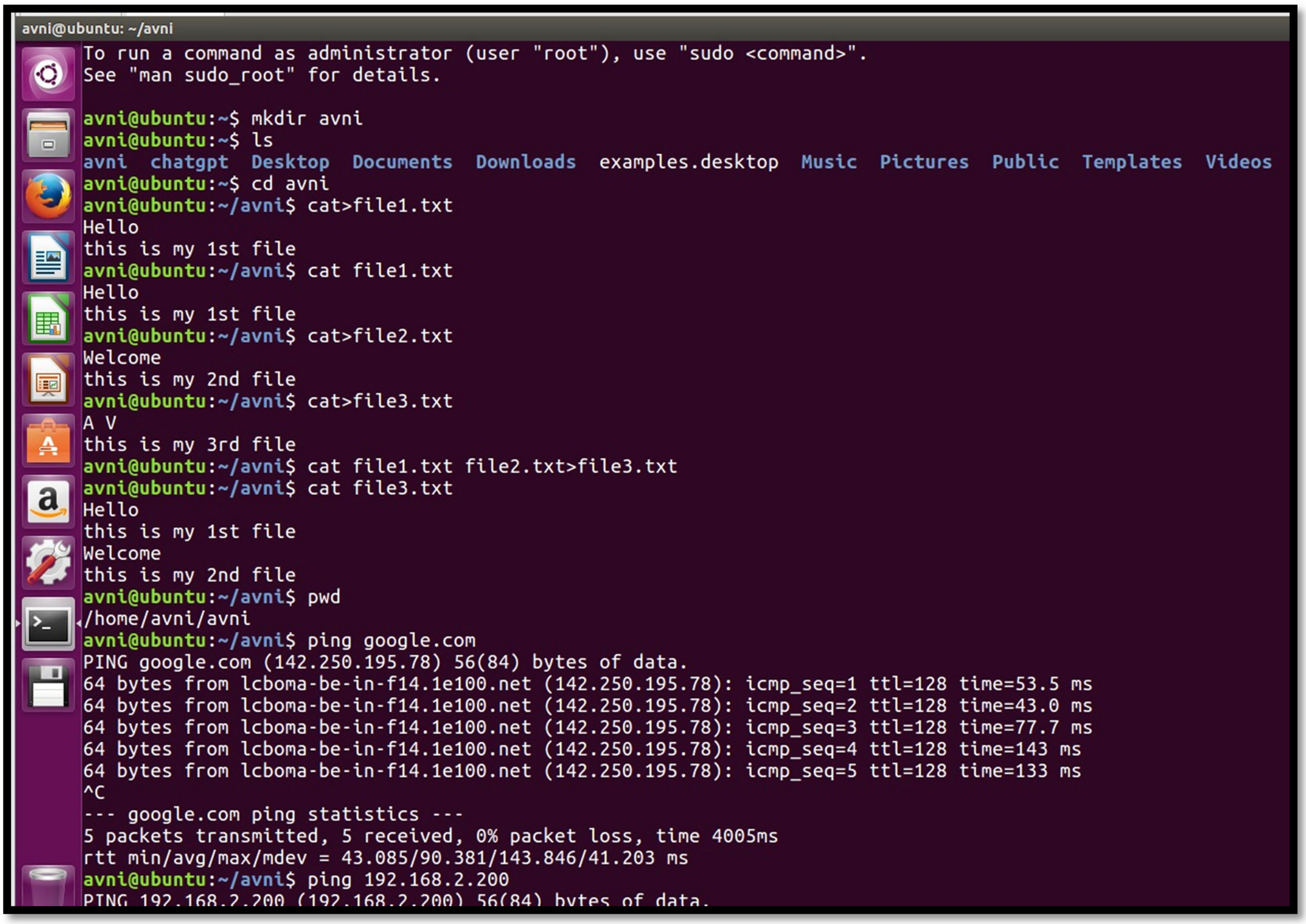
1. **What is the significance of the pwd command in a Linux environment?**
2. **Explain the function of the cp command and its common options.**
3. **How does chmod 700 affect file permissions, and what does each digit represent?**
4. **Describe the difference between head and tail commands in Linux.**
5. **What is the purpose of the grep command, and how is it used with regular expressions?**

❖ **References**:

https://ubuntu.com/tutorials/command-line-for-beginners#1-overview
https://www.geeksforgeeks.org/25-basic-ubuntu-commands/

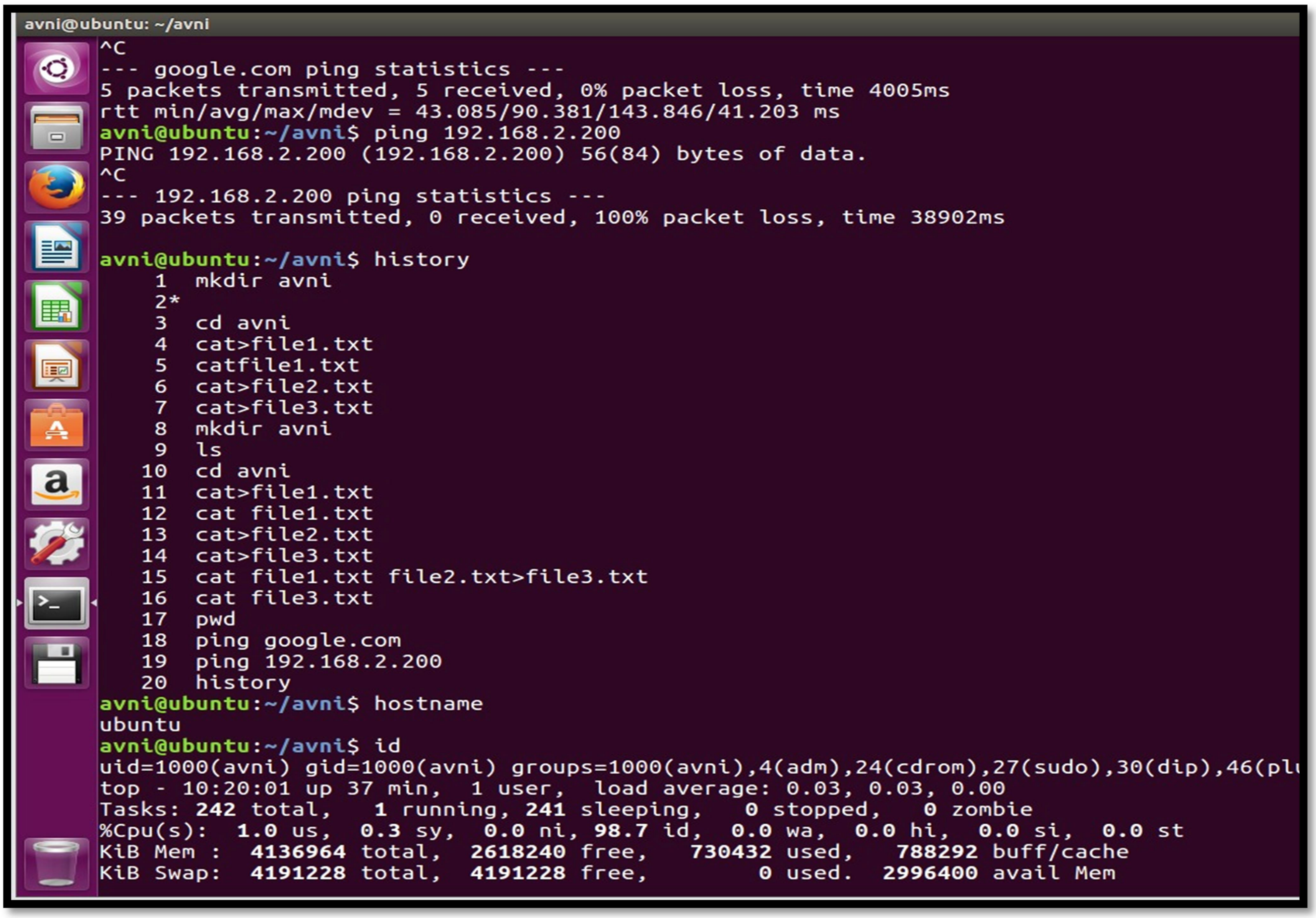
Date:___/___/2026

Signature
Course Coordinator
B.Tech CSE(AIML)
Sem: 4 / 2025-26



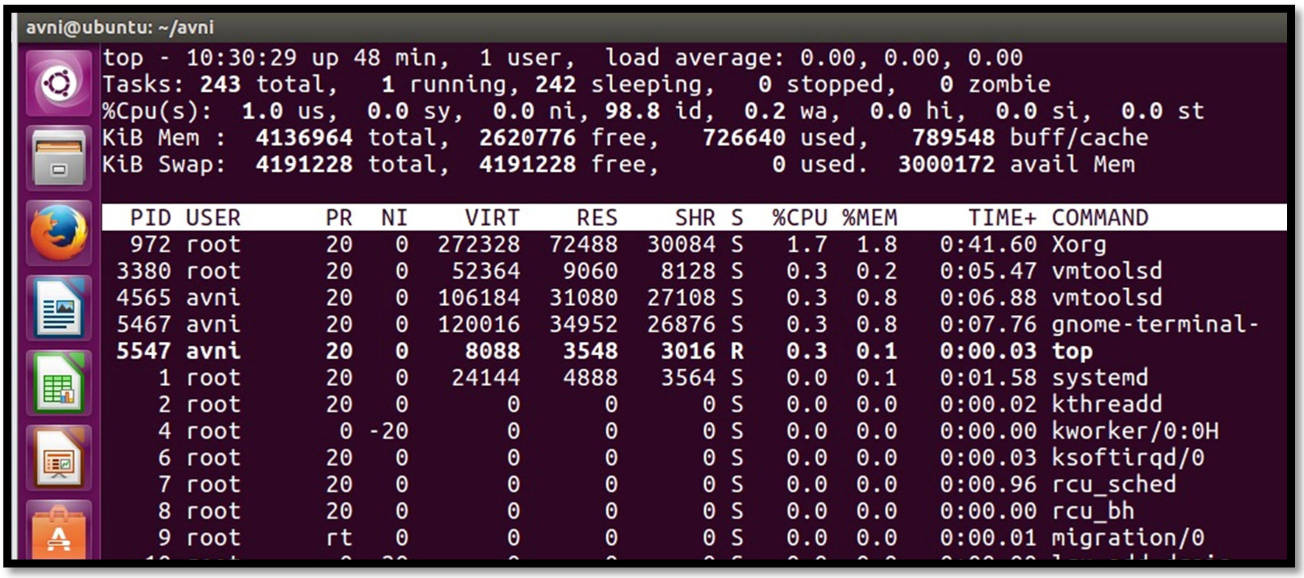
```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

avni@ubuntu:~$ mkdir avni
avni@ubuntu:~$ ls
avni  chatgpt  Desktop  Documents  Downloads  examples.desktop  Music  Pictures  Public  Templates  Videos
avni@ubuntu:~$ cd avni
avni@ubuntu:~/avni$ cat>file1.txt
Hello
this is my 1st file
avni@ubuntu:~/avni$ cat file1.txt
Hello
this is my 1st file
avni@ubuntu:~/avni$ cat>file2.txt
Welcome
this is my 2nd file
avni@ubuntu:~/avni$ cat>file3.txt
A V
this is my 3rd file
avni@ubuntu:~/avni$ cat file1.txt file2.txt>file3.txt
avni@ubuntu:~/avni$ cat file3.txt
Hello
this is my 1st file
Welcome
this is my 2nd file
avni@ubuntu:~/avni$ pwd
/home/avni/avni
avni@ubuntu:~/avni$ ping google.com
PING google.com (142.250.195.78) 56(84) bytes of data.
64 bytes from lcboma-be-in-f14.1e100.net (142.250.195.78): icmp_seq=1 ttl=128 time=53.5 ms
64 bytes from lcboma-be-in-f14.1e100.net (142.250.195.78): icmp_seq=2 ttl=128 time=43.0 ms
64 bytes from lcboma-be-in-f14.1e100.net (142.250.195.78): icmp_seq=3 ttl=128 time=77.7 ms
64 bytes from lcboma-be-in-f14.1e100.net (142.250.195.78): icmp_seq=4 ttl=128 time=143 ms
64 bytes from lcboma-be-in-f14.1e100.net (142.250.195.78): icmp_seq=5 ttl=128 time=133 ms
^C
--- google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 43.085/90.381/143.846/41.203 ms
avni@ubuntu:~/avni$ ping 192.168.2.200
PING 192.168.2.200 (192.168.2.200) 56(84) bytes of data.
```



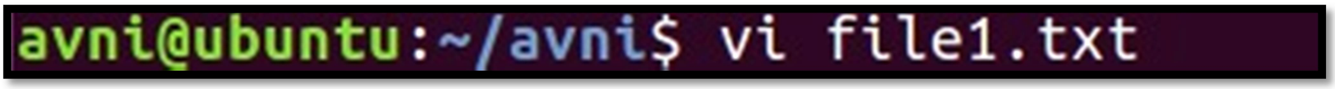
```
^C
--- google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 43.085/90.381/143.846/41.203 ms
avni@ubuntu:~/avni$ ping 192.168.2.200
PING 192.168.2.200 (192.168.2.200) 56(84) bytes of data.
^C
--- 192.168.2.200 ping statistics ---
39 packets transmitted, 0 received, 100% packet loss, time 38902ms

avni@ubuntu:~/avni$ history
    1  mkdir avni
    2*
    3  cd avni
    4  cat>file1.txt
    5  catfile1.txt
    6  cat>file2.txt
    7  cat>file3.txt
    8  mkdir avni
    9  ls
   10  cd avni
   11  cat>file1.txt
   12  cat file1.txt
   13  cat>file2.txt
   14  cat>file3.txt
   15  cat file1.txt file2.txt>file3.txt
   16  cat file3.txt
   17  pwd
   18  ping google.com
   19  ping 192.168.2.200
   20  history
avni@ubuntu:~/avni$ hostname
ubuntu
avni@ubuntu:~/avni$ id
uid=1000(avni) gid=1000(avni) groups=1000(avni),4(adm),24(cdrom),27(sudo),30(dip),46(pl
top - 10:20:01 up 37 min,  1 user,  load average: 0.03, 0.03, 0.00
Tasks: 242 total,   1 running, 241 sleeping,   0 stopped,   0 zombie
%Cpu(s):  1.0 us,  0.3 sy,  0.0 ni, 98.7 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem :  4136964 total,  2618240 free,   730432 used,   788292 buff/cache
KiB Swap:  4191228 total,  4191228 free,        0 used.  2996400 avail Mem
```

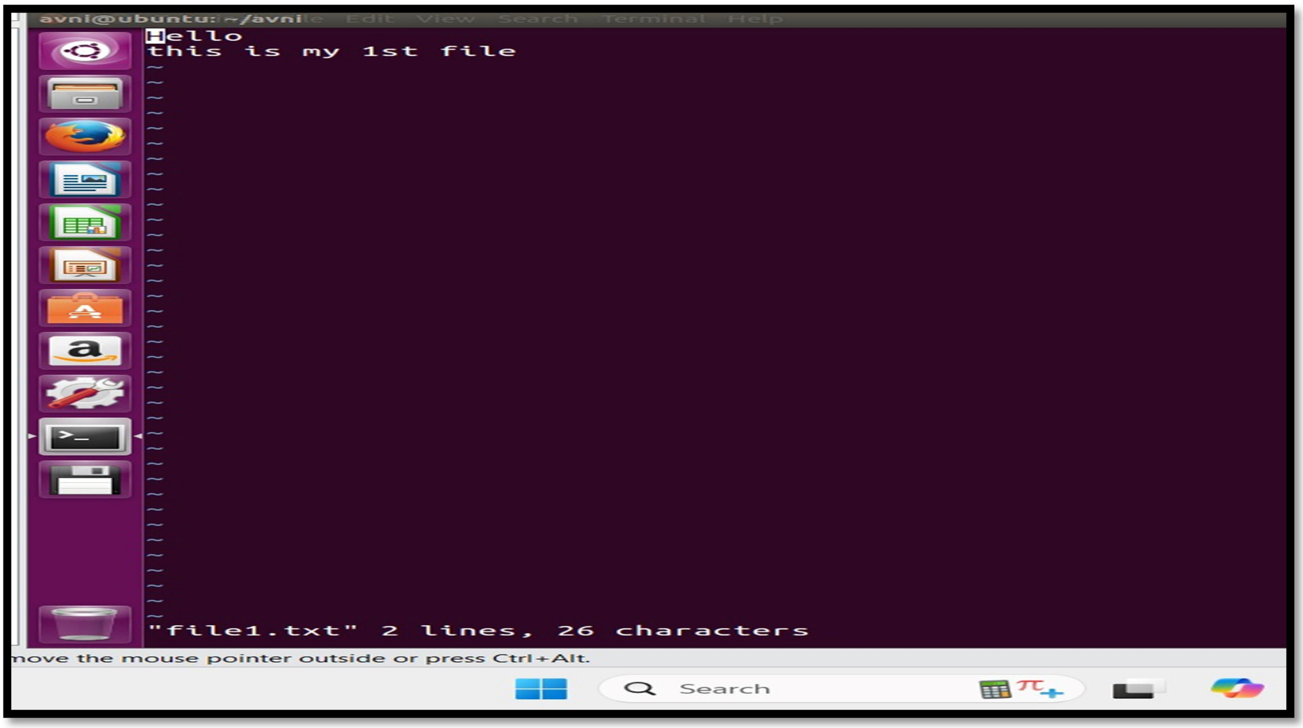


```
top - 10:30:29 up 48 min,  1 user,  load average: 0.00, 0.00, 0.00
Tasks: 243 total,   1 running, 242 sleeping,   0 stopped,   0 zombie
%Cpu(s):  1.0 us,  0.0 sy,  0.0 ni, 98.8 id,  0.2 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem :  4136964 total,  2620776 free,   726640 used,   789548 buff/cache
KiB Swap:  4191228 total,  4191228 free,        0 used.  3000172 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
  972 root      20   0  272328  72488  30084 S   1.7  1.8   0:41.60 Xorg
 3380 root      20   0   52364   9060   8128 S   0.3  0.2   0:05.47 vmtoolsd
 4565 avni      20   0  106184  31080  27108 S   0.3  0.8   0:06.88 vmtoolsd
 5467 avni      20   0  120016  34952  26876 S   0.3  0.8   0:07.76 gnome-terminal-
 5547 avni      20   0    8088   3548   3016 R   0.3  0.1   0:00.03 top
    1 root      20   0   24144   4888   3564 S   0.0  0.1   0:01.58 systemd
    2 root      20   0       0      0      0 S   0.0  0.0   0:00.02 kthreadd
    4 root       0 -20       0      0      0 S   0.0  0.0   0:00.00 kworker/0:0H
    6 root      20   0       0      0      0 S   0.0  0.0   0:00.03 ksoftirqd/0
    7 root      20   0       0      0      0 S   0.0  0.0   0:00.96 rcu_sched
    8 root      20   0       0      0      0 S   0.0  0.0   0:00.00 rcu_bh
    9 root      rt   0       0      0      0 S   0.0  0.0   0:00.01 migration/0
```

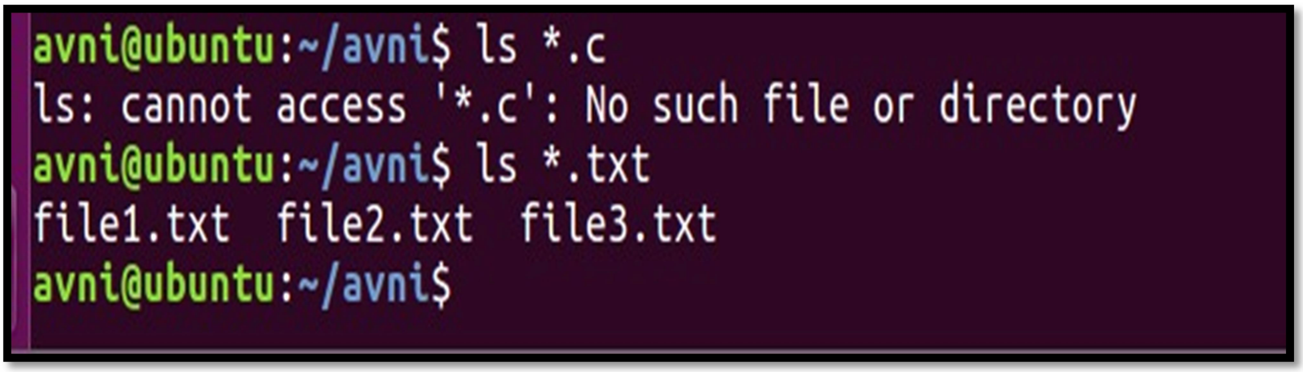


```
avni@ubuntu:~/avni$ vi file1.txt
```



```
Hello
this is my 1st file
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"file1.txt" 2 lines, 26 characters
```

move the mouse pointer outside or press Ctrl+Alt.



```
avni@ubuntu:~/avni$ ls *.c
ls: cannot access '*.c': No such file or directory
avni@ubuntu:~/avni$ ls *.txt
file1.txt  file2.txt  file3.txt
avni@ubuntu:~/avni$
```