K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)
Department of Computer Engineering

Batch: A2	Roll No.: 16010121045

Experiment No. 01

Grade: AA / AB / BB / BC / CC / CD /DD

Signature of the Staff In-charge with date

TITLE: Exploring basic Commands of UNIX: Shell, Processes, Files

---

AIM: To Explore basic commands for handling File system under Unix/Linux using shell scripts.(Creating groups, chown , chmod , directory name, tty , diff, umask).

Expected Outcome of Experiment:

CO 1. To introduce basic concepts and functions of operating systems.

Books/ Journals/ Websites referred:

1.   **Silberschatz A., Galvin P., Gagne G. "Operating Systems Principles", Willey Eight edition.**
2.   **Achyut S. Godbole , Atul Kahate "Operating Systems", McGraw Hill Third Edition.**
3.   **Sumitabha Das " UNIX Concepts & Applications", McGraw Hill Second Edition.**

Pre Lab/ Prior Concepts:

An operating system (OS) is a resource manager. It takes the form of a set of software routines that allow users and application programs to access system resources (e.g. the CPU, memory, disks, modems, printers network cards etc.) in safe efficient and abstract way.

•   The operating system kernel is in direct control of the underlying hardware. The kernel provides low-level device, memory and processor management functions (e.g. dealing with interrupts from hardware devices, sharing the processor among multiple programs, allocating memory for programs etc.)

•   Basic hardware-independent kernel services are exposed to higher-level programs through a library of system calls (e.g. services to create a file, begin execution of a program, or open a logical network connection to another computer).

Department of Computer Engineering

- Application programs (e.g. word processors, spreadsheets) and system utility programs (simple but useful application programs that come with the operating system, e.g. programs which find text inside a group of files) make use of system calls. Applications and system utilities are launched using a shell (a textual command line interface) or a graphical user interface that provides direct user interaction.

Operating systems can be distinguished from one another by the system calls, system utilities and user interface they provide, as well as by the resource scheduling policies implemented by the kernel.

UNIX has been a popular OS for more than two decades because of its multi-user, multi-tasking environment, stability, portability and powerful networking capabilities.

Linux is a free open source UNIX OS for PCs.

Linux has all of the components of a typical OS :

- ### Kernel

The Linux kernel includes device driver support for a large number of PC hardware devices (graphics cards, network cards, hard disks etc.), advanced processor and memory management features, and support for many different types of file systems. In terms of the services that it provides to application programs and system utilities, the kernel implements most BSD and SYSV system calls, as well as the system calls described in the POSIX.1 specification.

The kernel (in raw binary form that is loaded directly into memory at system startup time) is typically found in the file /boot/vmlinuz, while the source files can usually be found in /usr/src/linux.

- ### Shells and GUIs

Linux supports two forms of command input: through textual command line shells similar to those found on most UNIX systems (e.g. sh - the Bourne shell, bash - the Bourne again shell and csh - the C shell) and through graphical interfaces (GUIs) such as the KDE and GNOME window managers.

- ### System Utilities

Virtually every system utility that you would expect to find on standard implementations of UNIX has been ported to Linux. This includes commands such as ls, cp, grep, awk, sed, bc, wc, more, and so on. These system utilities are designed to be powerful tools that do a single task extremely well (e.g. grep finds text inside files while wc counts the number of words, lines and bytes inside a file). Users can often solve problems by interconnecting these tools instead of writing a large monolithic application program.

- ### Application programs

Linux distributions typically come with several useful application programs as standard. Examples include the emacseditor, xv (an image viewer), gcc (a C compiler), g++ (a C++ compiler), xfig (a drawing package), latex (a powerful typesetting language) and soffice (StarOffice, which is an MS-Office style clone that can read and write Word, Excel and PowerPoint files).

Description of Commands and options:

DOS commands: Attrib, dir, at, chkdsk, shutdown, tree, create a batch file, output and input redirection
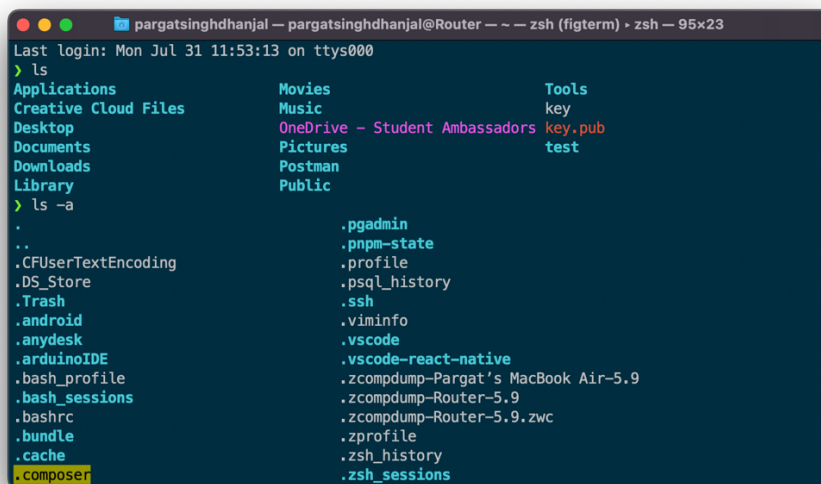
Windows utilities: msconfig, defragmenter, performance monitor, task manager, registry editor, event viewer, process explorer

Unix Commands:

1)   Unix file operations: ls, cp, rm , mv, chmod, chown ,chgrp
2)   Text file operations in Unix :   cat , more , less , head, tail , grep
3)   Unix directory management commands :   cd, pwd , ln, mkdir, rmdir
4)   Unix system status commands: hostname, w, uname
5)   Process management: ps, top, kill
6)   Unix users commands: whoami , id, groups, passwd , who, last

Implementation details:

ls: List files and directories in the current directory.
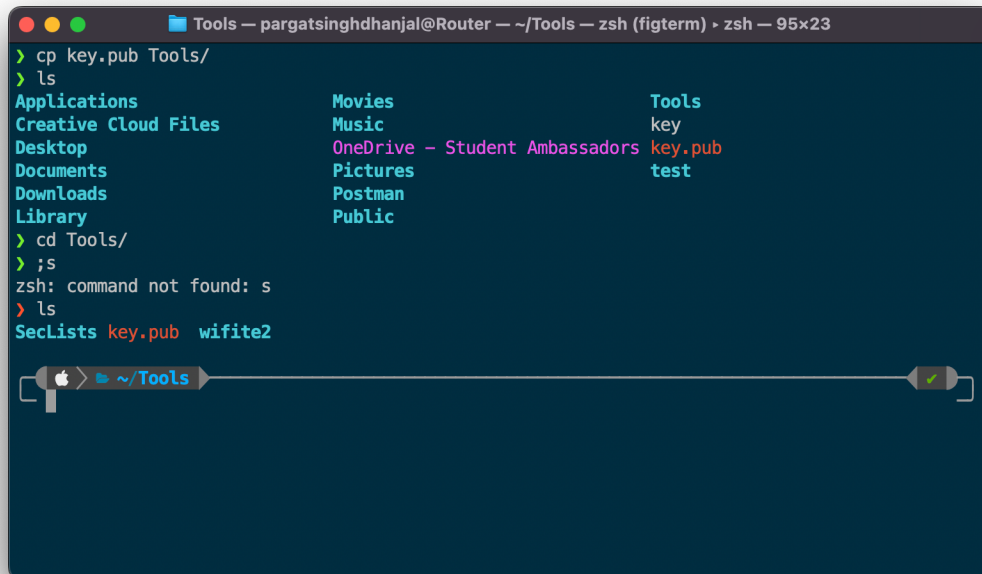
Department of Computer Engineering

cp: Copy files or directories from one location to another.
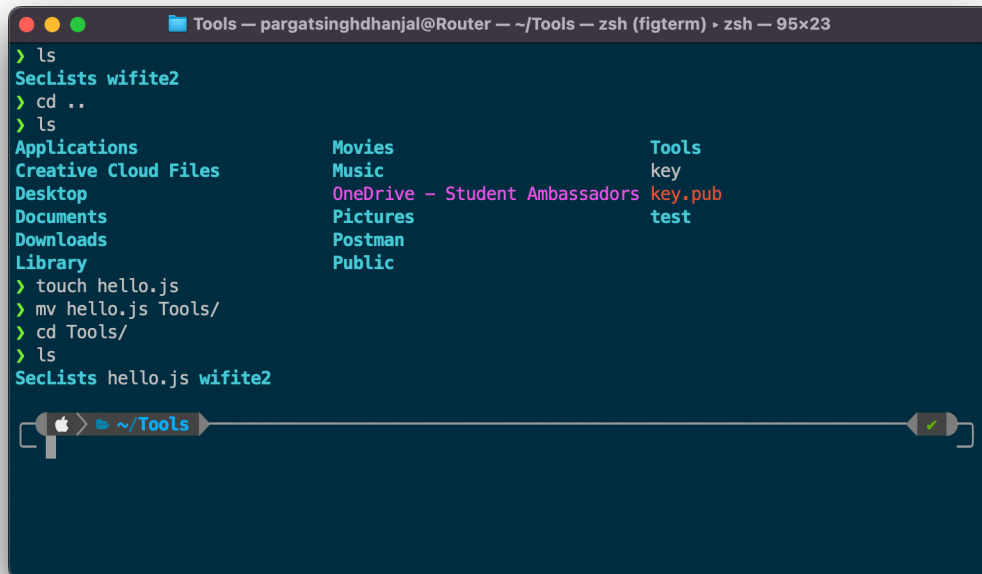

```
> cp key.pub Tools/
> ls
Applications              Movies                    Tools
Creative Cloud Files      Music                     key
Desktop                   OneDrive — Student Ambassadors key.pub
Documents                 Pictures                  test
Downloads                 Postman
Library                   Public
> cd Tools/
> ;s
zsh: command not found: s
> ls
SecLists key.pub  wifite2
```
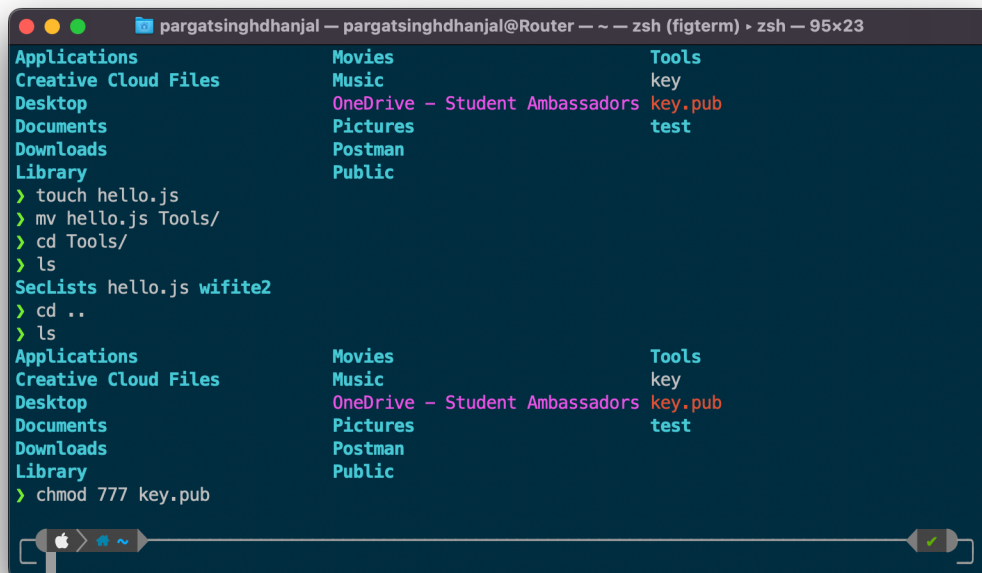
rm: Remove (delete) files or directories.


```
> ls
SecLists key.pub  wifite2
> rm key.pub
> ls
SecLists wifite2
```

mv: Move or rename files or directories.

chmod: Change file permissions (e.g., read, write, execute) for users.

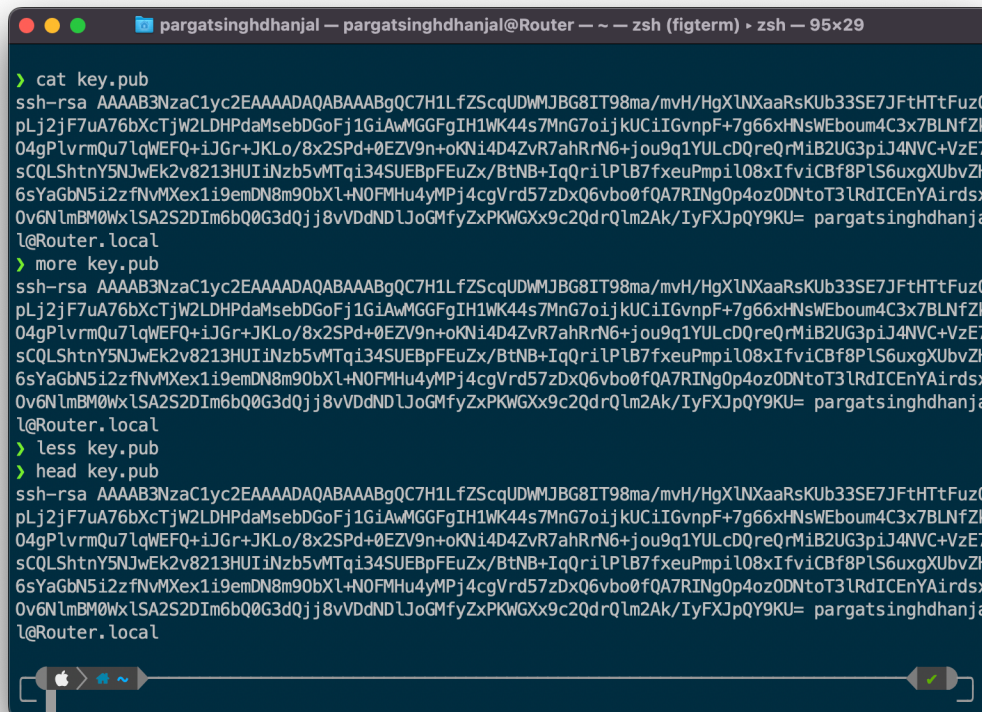cat: Concatenate and display the contents of a file.

more: Display a file one screen at a time.

less: Display a file with backward navigation support.

head: Display the beginning lines of a file.

tail: Display the ending lines of a file.

grep: Search for a pattern in a file or stream and print matching lines.

SOMAIYA
VIDYAVIHAR UNIVERSITY
K J Somaiya College of Engineering

K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)
Department of Computer Engineering
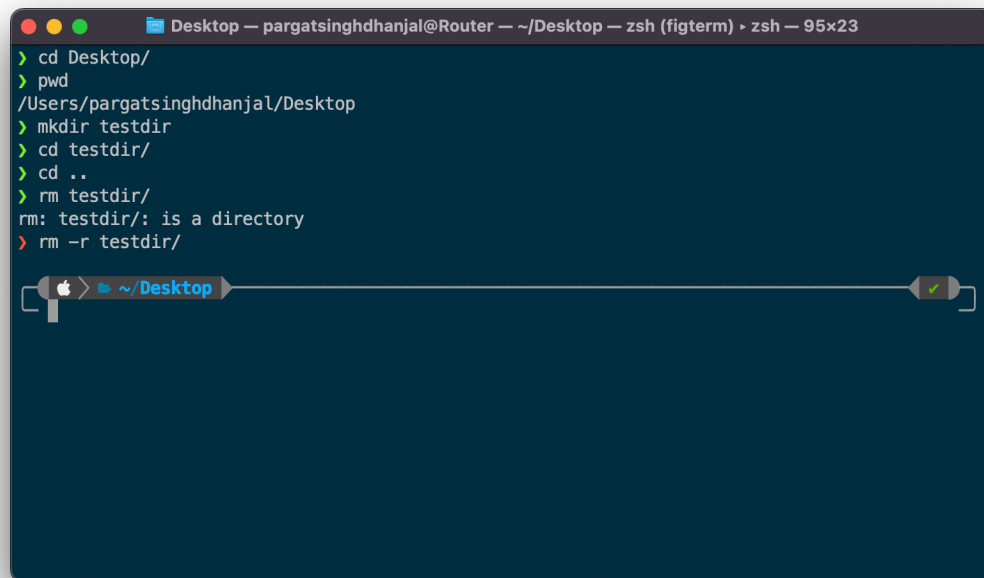
Somaiya
TRUST

cd: Change the current working directory.

pwd: Print the current working directory.

ln: Create hard or symbolic links to files or directories.

mkdir: Create a new directory.

rmdir: Remove an empty directory.

```
Desktop — pargatsinghdhanjal@Router — ~/Desktop — zsh (figterm) ‣ zsh — 95×23
❯ cd Desktop/
❯ pwd
/Users/pargatsinghdhanjal/Desktop
❯ mkdir testdir
❯ cd testdir/
❯ cd ..
❯ rm testdir/
rm: testdir/: is a directory
❯ rm -r testdir/

     ~/Desktop                                                              ✓
```

hostname: Display or set the system's host name.

w: Show who is logged on and what they are doing.

uname: Print system information (e.g., kernel name, node name).
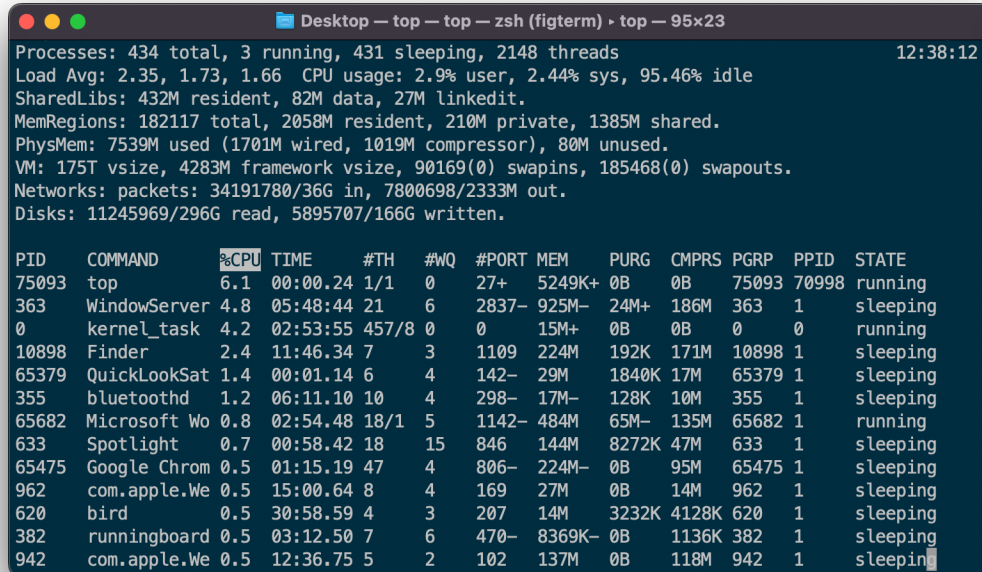
ps: Display information about running processes.

```
Desktop — pargatsinghdhanjal@Router — ~/Desktop — zsh (figterm) ‣ zsh — 95×23
❯ hostname
Router.local
❯ w
12:36  up 4 days, 21:58, 2 users, load averages: 1.49 1.58 1.61
USER     TTY      FROM              LOGIN@  IDLE WHAT
pargatsinghdhanjal console  -                Wed14   4days  -
pargatsinghdhanjal s000     -                12:25      - zsh (figterm)
❯ uname
Darwin
❯ ps
  PID TTY          TIME CMD
70988 ttys000    0:05.30 zsh (figterm)
70998 ttys001    0:03.19 /bin/zsh --login
71022 ttys001    0:00.00 /bin/zsh --login
71057 ttys001    0:00.00 /bin/zsh --login
71058 ttys001    0:00.01 /bin/zsh --login
71060 ttys001    0:00.03 /Users/pargatsinghdhanjal/.cache/gitstatus/gitstatusd-darwin-arm64 -G
```

Department of Computer Engineering

top: Monitor and display dynamic system processes and resource usage.

kill: Send a signal to terminate a process or send specific signals to control a process.

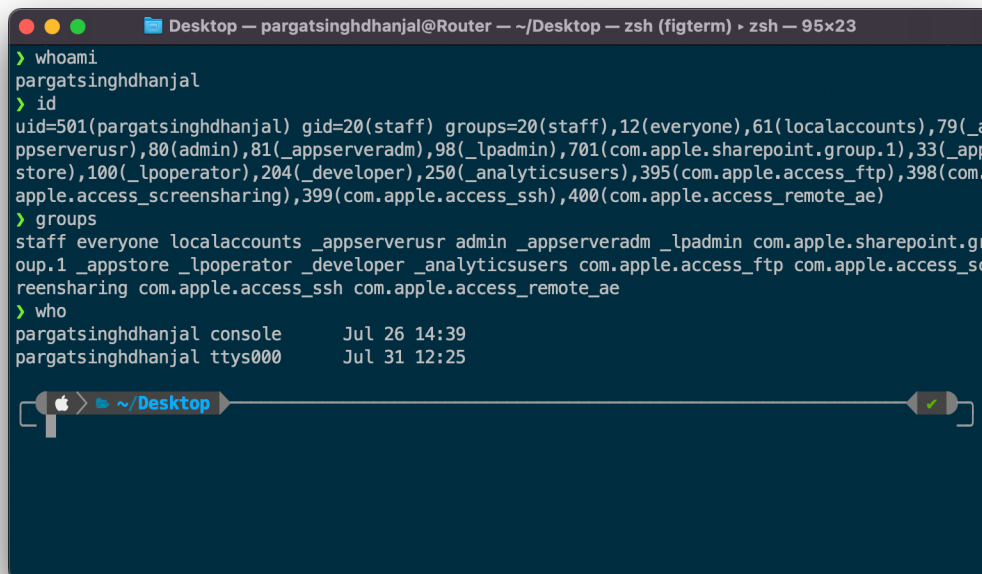whoami: Print the effective user ID.

id: Display user and group ID information.

groups: Display the groups a user belongs to.

passwd: Change a user's password.

who: Show who is logged on.

last: Display recent login activity of users.

Department of Computer Engineering

Conclusion:

Explored basic Commands of UNIX: Shell, Processes, Files.

Post Lab Descriptive Questions

1.   Explain how do you read and interpret syntax of any OS command.

Reading and interpreting the syntax of any OS command involves understanding the structure and format of the command and its parameters. In most operating systems, commands follow a specific pattern: the command itself is typically followed by options or flags that modify its behavior, and these options are often preceded by a hyphen or double hyphen. Additionally, the command may require arguments that provide specific inputs or targets for the operation. To interpret a command, one must refer to the OS documentation or use the built-in help command (e.g., "man" in Unix-like systems or "help" in Windows) to understand the purpose and proper usage of each command, along with its available options and argument syntax.

2.   Explain different functions of the operating systems.

Operating systems serve various functions to manage computer resources efficiently and provide a user-friendly environment. Some key functions include process management, memory management, file system management, device management, and user interface. Process management involves scheduling and controlling processes to ensure efficient utilization of the CPU. Memory management handles the allocation and deallocation of memory to running processes. File system management handles file organization and storage. Device management ensures proper communication with hardware devices like printers and disk drives. User interface allows users to interact with the system through a graphical or command-line interface, making it easier to access and manage resources.

3.   What are the default permissions assigned by Unix for Directory.

In Unix-like systems, the default permissions assigned for directories are usually "rwxr-xr-x" or 755 in octal notation. This means the owner of the directory has read (r), write (w), and execute (x) permissions, while the group and others have only read and execute permissions. The read permission allows viewing the contents of the directory, write permission allows creating or deleting files within the directory, and execute permission enables access to the contents of the directory, such as listing its files.

4.   Give the difference between DOS and WINDOWS.

DOS (Disk Operating System) and Windows are both operating systems, but they differ significantly in terms of architecture, features, and capabilities. DOS was a single-tasking, command-line-based operating system developed by Microsoft, primarily used during the early days of personal computing. It lacked multitasking support and a graphical user interface (GUI). Windows, on the other hand, is a multitasking, multi-user operating system with a GUI. It provides a more sophisticated and user-friendly environment, allowing users to run multiple programs simultaneously and access various hardware

Department of Computer Engineering

devices more efficiently. Windows also supports networking, making it easier to connect computers and share resources.

5.   Explain Booting Process.

The booting process is the sequence of events that occur when a computer is powered on, leading to the loading of the operating system. When the computer is turned on, the Basic Input/Output System (BIOS) or Unified Extensible Firmware Interface (UEFI) performs a Power-On Self-Test (POST) to check the hardware's integrity. Afterward, the BIOS/UEFI searches for the boot loader in the system's boot devices (usually the hard drive). The boot loader (e.g., GRUB for Linux or NTLDR for older versions of Windows) then loads the kernel or core components of the operating system into memory. The kernel takes control and initializes necessary drivers and services. Finally, the user interface or desktop environment is presented, and the computer is ready for user interaction.

Date: _____   **Signature of faculty in-charge**

Department of Computer Engineering