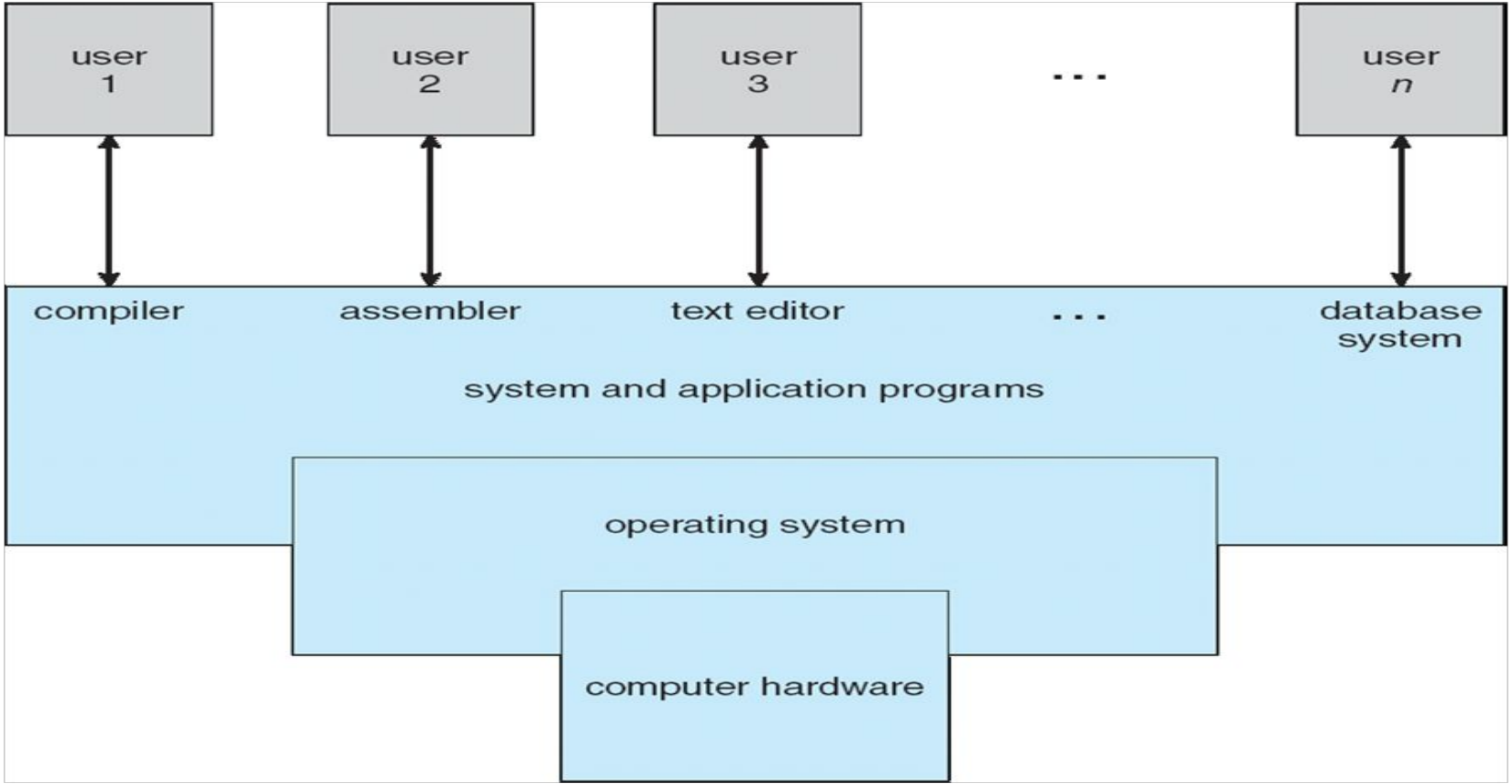
Module-1

Fundamentals of Operating System

Minakshi Retharekar

Components of a Computer System



Computer System Structure

Computer system can be divided into four components:

- **Hardware** – provides basic computing resources.

  CPU, memory, I/O devices

- **Operating system**

  Controls and coordinates use of hardware among various applications and users.

- **Application programs** – define the ways in which the system resources are used to solve the computing problems of the users.

  Word processors, compilers, web browsers, database systems, video games

- **Users**

  People, machines, other computers.

•Computer System = Hardware + Software

•Software = Application Software + System Software(OS)

- Application Software: application software is a computer software package that performs a specific function for a user, or in some cases, for another application

For Ex. office suites, graphics software, databases and database management programs, web browsers, word processors, software development tools, image editors and communication platforms

- System Software:These software programs are designed to run a computer's application programs and hardware. System software coordinates the activities and functions of the hardware and software.

•An Operating System is a system Software that acts as an intermediary/interface between a user of a computer and the computer hardware.

•Operating system goals:

  - Execute user programs and make solving user problems easier
  - Make the computer system convenient to use
  - Use the computer hardware in an efficient manner

Operating System

- A program that acts as an intermediary between a user of a computer and the computer hardware.

❖ **OS is a resource allocator**

➢ Manages all resources

➢ Decides between conflicting requests for efficient and fair resource use

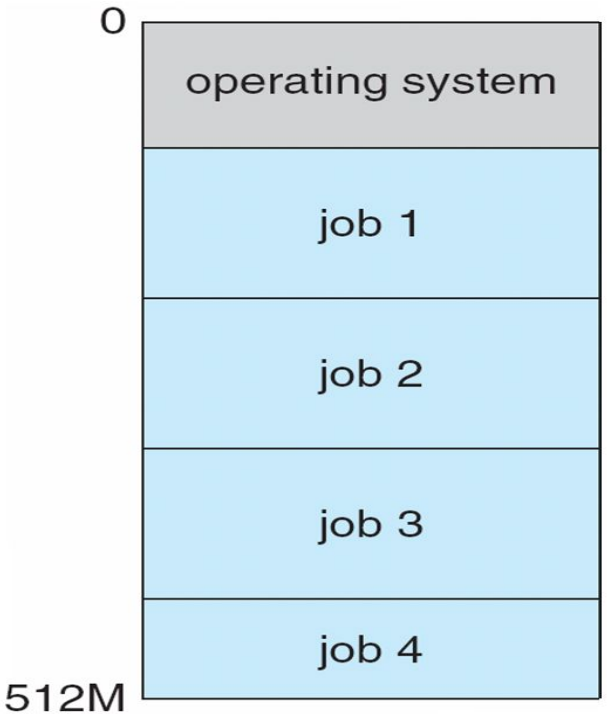
❖ **OS is a control program**

Controls execution of programs to prevent errors and improper use of the computer

Operating System Structure

- **Multiprogramming** needed for efficiency
  - Single user cannot keep CPU and I/O devices busy at all times
  - Multiprogramming organizes jobs (code and data) so CPU always has one to execute
  - A subset of total jobs in system is kept in memory
  - One job selected and run via **job scheduling**
  - When it has to wait (for I/O for example), OS switches to another job
- **Timesharing (multitasking)** is logical extension in which CPU switches jobs so frequently that users can interact with each job while it is running, creating **interactive** computing
  - **Response time** should be < 1 second
  - Each user has at least one program executing in memory □**process**
  - If several jobs ready to run at the same time □ **CPU scheduling**
  - If processes don't fit in memory, **swapping** moves them in and out to run
  - **Virtual memory** allows execution of processes not completely in memory

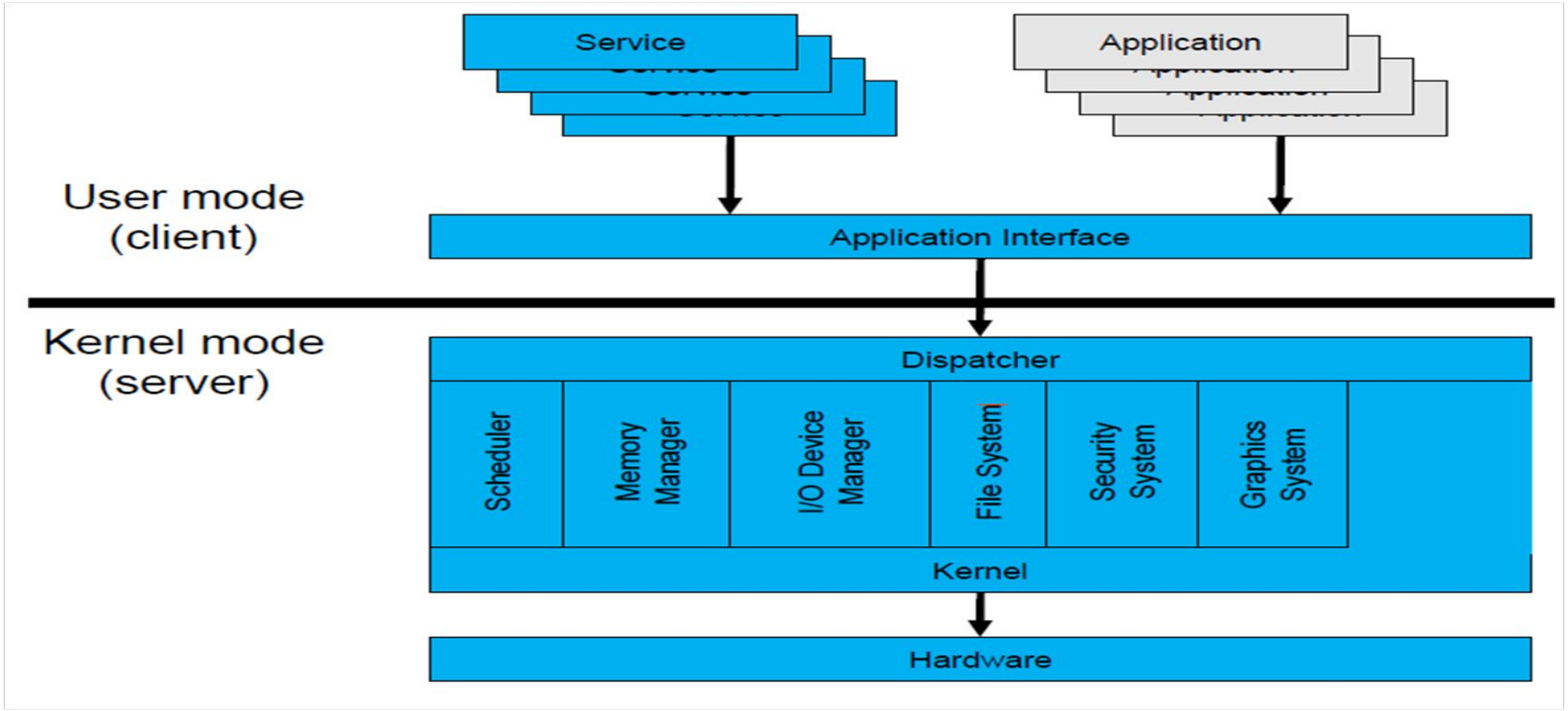
Memory Layout for Multiprogrammed System



Operating-System Operations

- Interrupt driven by hardware
- Software error or request creates **exception** or **trap**
  - Division by zero, request for operating system service
- Other process problems include infinite loop, processes modifying each other or the operating system
- **Dual-mode** operation allows OS to protect itself and other system components
  - **User mode** and **kernel mode**
  - **Mode bit** provided by hardware
    - Provides ability to distinguish when system is running user code or kernel code
    - Some instructions designated as **privileged**, only executable in kernel mode
    - System call changes mode to kernel, return from call resets it to user

Operating System Mode



Operating System Mode

- The User Mode is concerned with the actual interface between the user and the system.
  -It controls things like running applications and accessing files.

- The Kernel Mode is concerned with everything running in the background.
  -It controls things like accessing system resources, controlling hardware functions and processing program instructions.

- System calls are used to change mode from User to Kernel.

Kernel

- Kernel is a **software code that reside in central core of OS**. It has complete control over system.
- When operating system boots, kernel is first part of OS to load in main memory.
- **Objectives of Kernel :**
  - To establish communication between user level application and hardware.
  - To decide state of incoming processes.
  - To control disk management.
  - To control memory management.
  - To control task management.

- Kernel includes:-

    1. Scheduler: It allocates the Kernel's processing time to various processes.

    2. Supervisor: It grants permission to use computer system resources to each process.

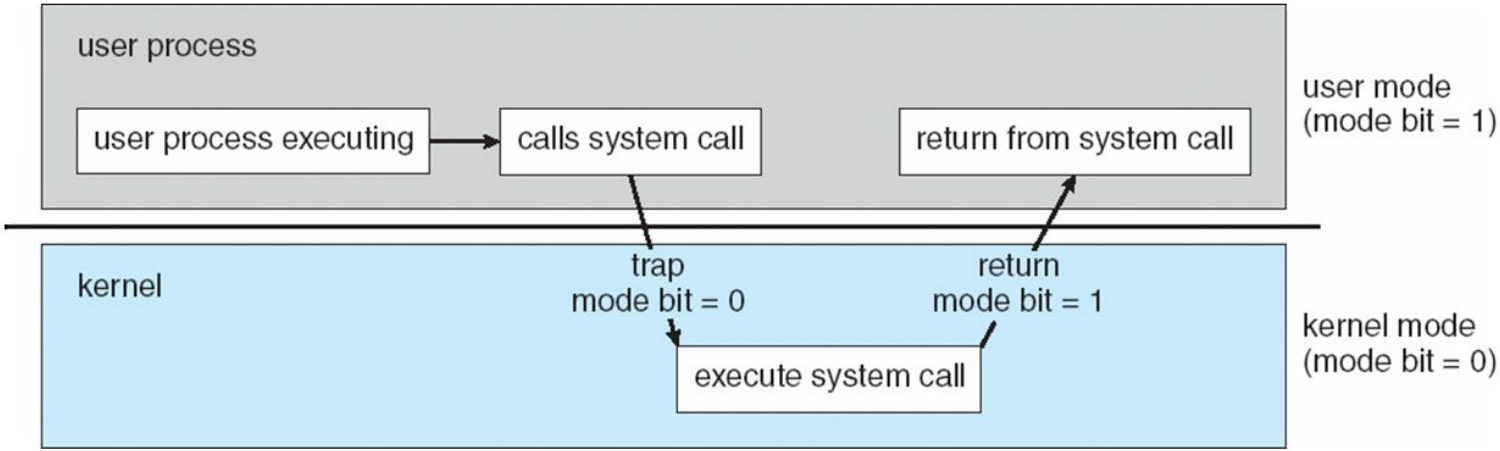
    3. Interrupt handler : It handles all requests from the various hardware devices which compete for kernel services.

    4. Memory manager : allocates space in memory for all users of kernel service.

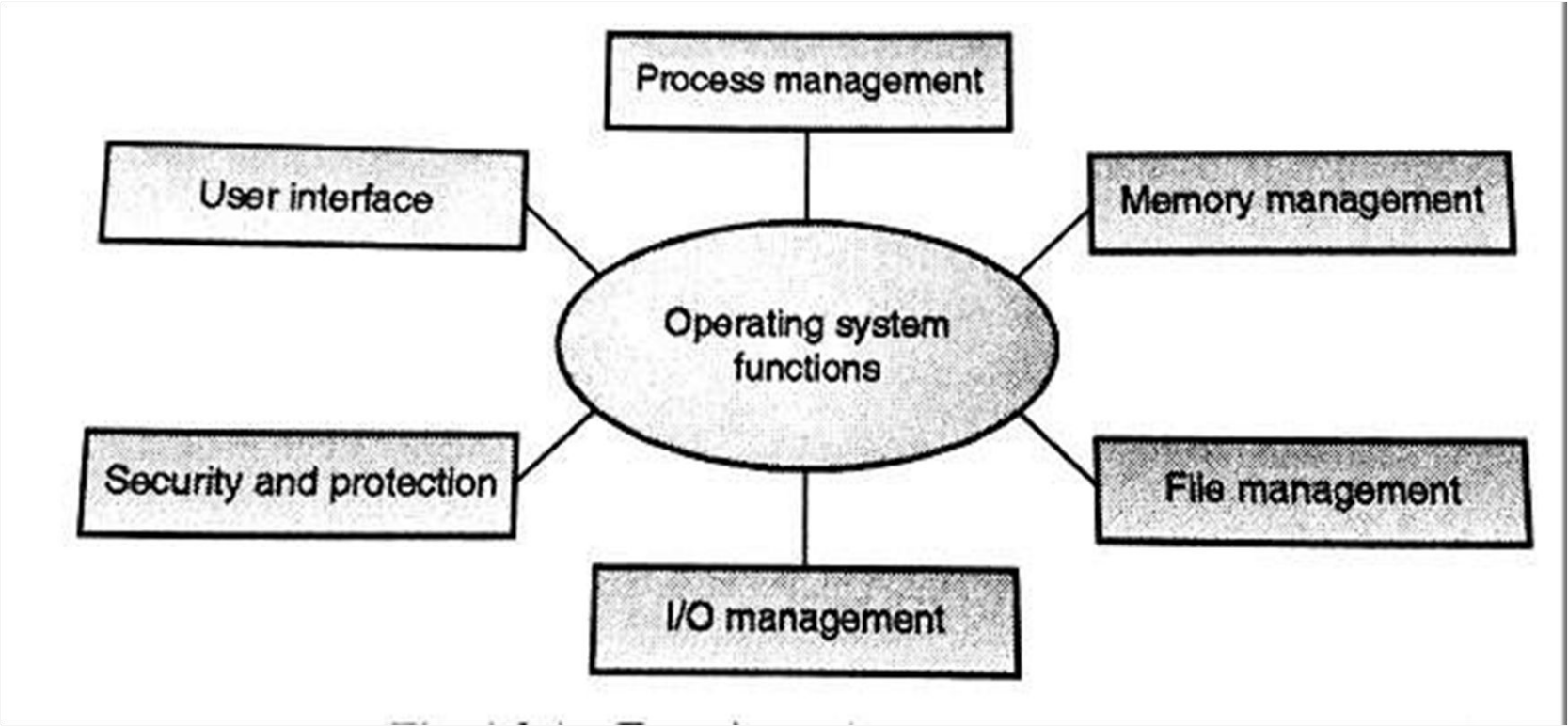
- kernel provides services for process management, file management, I/O management, memory management.
- System calls are used to provide these type of services.

Transition from User to Kernel Mode

- Timer to prevent infinite loop / process hogging resources
  - Set interrupt after specific period
  - Operating system decrements counter
  - When counter zero generate an interrupt
  - Set up before scheduling process to regain control or terminate program that exceeds allotted time



Operating System Functions



1. Memory Management

- All data in memory before and after processing.
- All instructions in memory in order to execute
- Memory management determines what is in memory when

  Optimizing CPU utilization and computer response to users
- Memory management activities

  Keeping track of which parts of memory are currently being used and by whom

  Deciding which processes (or parts thereof) and data to move into and out of memory

  Allocating and deallocating memory space as needed

2. Process Management

- **A *process* is a program in execution.**
- A process needs certain resources, including CPU time, memory, files, and I/O devices to accomplish its task.
- Simultaneous execution leads to multiple processes. Hence creation, execution and termination of a process are the most basic functionality of an OS
- If processes are dependent, than they may try to share same resources. thus task of process synchronization comes to the picture.
- If processes are independent, than a due care needs to be taken to avoid their overlapping in memory area.
- Based on priority, it is important to allow more important processes to execute first than others.

3. File Management

- A file is a collection of related information defined by its creator.
- *File systems provide the conventions for the encoding, storage and management of data on a storage device such as a hard disk.*
  - FAT12 (floppy disks)
  - FAT16 (DOS and older versions of Windows)
  - FAT32 (older versions of Windows)
  - NTFS (newer versions of Windows)
  - EXT3 (Unix/Linux)
  - HFS+ (Max OS X)
- The operating system is responsible for the following activities in connections with file management:
  - File creation and deletion.
  - Directory creation and deletion.
  - Support of primitives for manipulating files and directories.
  - Mapping files onto secondary storage.
  - File backup on stable (nonvolatile) storage media.

4. Device Management

- *Device controllers are components on the motherboard (or on expansion cards) that act as an interface between the CPU and the actual device.*
- *Device drivers, which are the operating system software components that interact with the devices controllers.*
- A special device (inside CPU) called the Interrupt Controller handles the task of receiving interrupt requests and prioritizes them to be forwarded to the processor.
- Deadlocks can occur when two (or more) processes have control of different I/O resources that are needed by the other processes, and they are unwilling to give up control of the device.
- It performs the following activities for device management.
  - Keeps tracks of all devices connected to system.
  - Designates a program responsible for every device known as Input/output controller.
  - Decides which process gets access to a certain device and for how long.
  - Allocates devices in an effective and efficient way.
  - Deallocates devices when they are no longer required.

5. Security & Protection

- The operating system uses password protection to protect user data and similar other techniques.
- It also prevents unauthorized access to programs and user data by assigning access right permission to files and directories.
- The owners of information stored in a multiuser or networked computer system may want to control use of that information, concurrent processes should not interfere with each other.

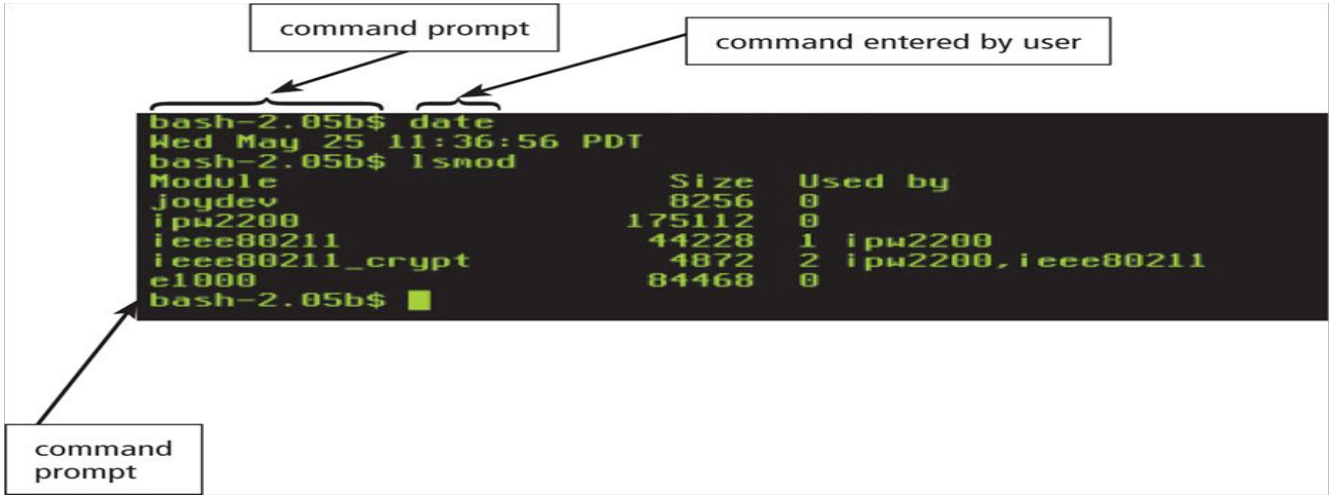
6. User Interface Mechanism

- A **user interface (UI)** controls how you enter data and instructions and how information is displayed on the screen
- There are two types of user interfaces

1. Command Line Interface
2. Graphical user Interface

1. Command-line interface

- In a command-line interface, a user types commands represented by short keywords or abbreviations or presses special keys on the keyboard to enter data and instructions.



2. Graphical User Interface

•With a graphical user interface (GUI), you interact with menus and visual images.

System Call

System call is the programmatic way in which a computer program/user application requests a service from the kernel of the operating system on which it is executed.

Application program is just a user-process. Due to security reasons , user applications are not given access to privileged resources(the ones controlled by OS).

When they need to **do any I/O** or have **some more memory** or wait for **signal/interrupt,** it requests operating system to facilitate all these. This **request is made through System Call.**

System calls are also called **software-interrupts.**