OPERATING SYSTEMS

LECTURE # 1

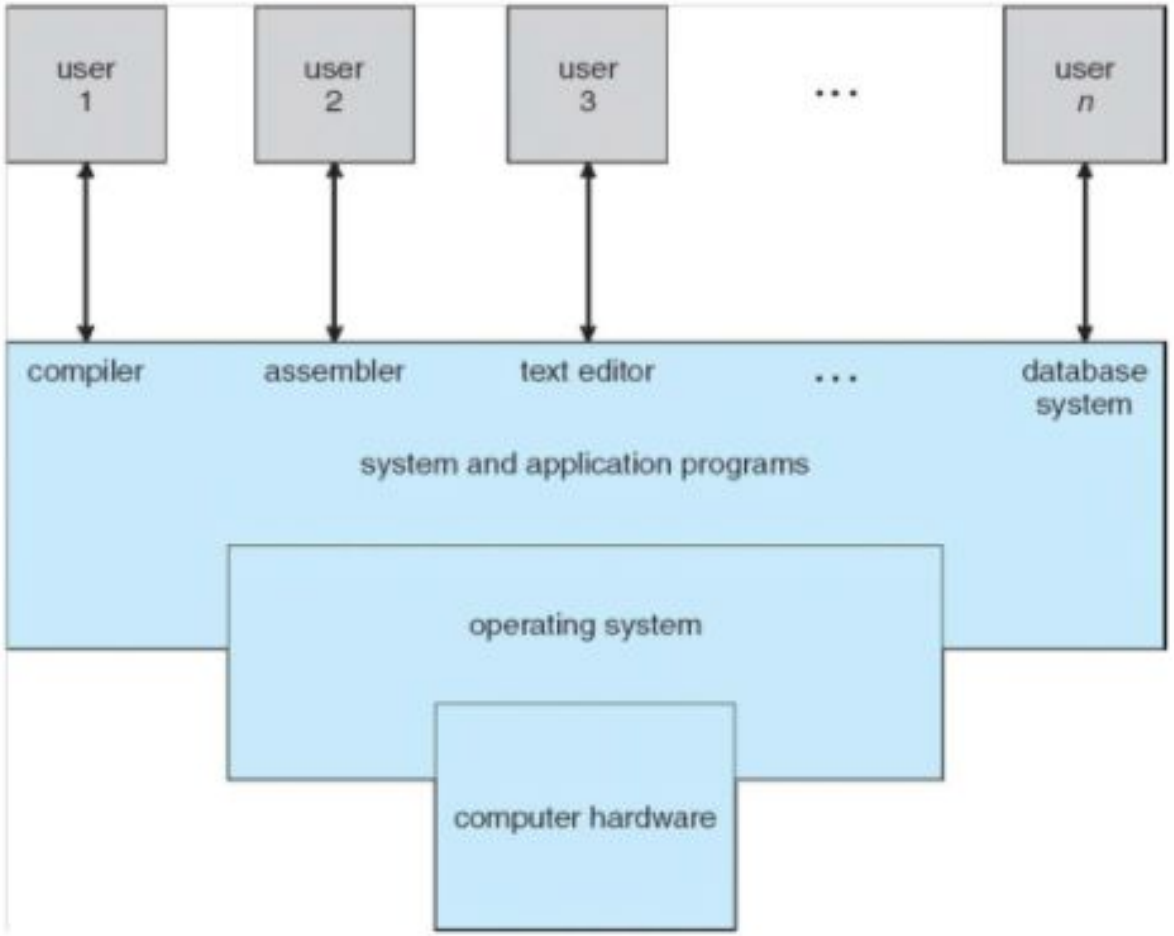
RAZI UDDIN

PURPOSE OF COMPUTER SYSTEM

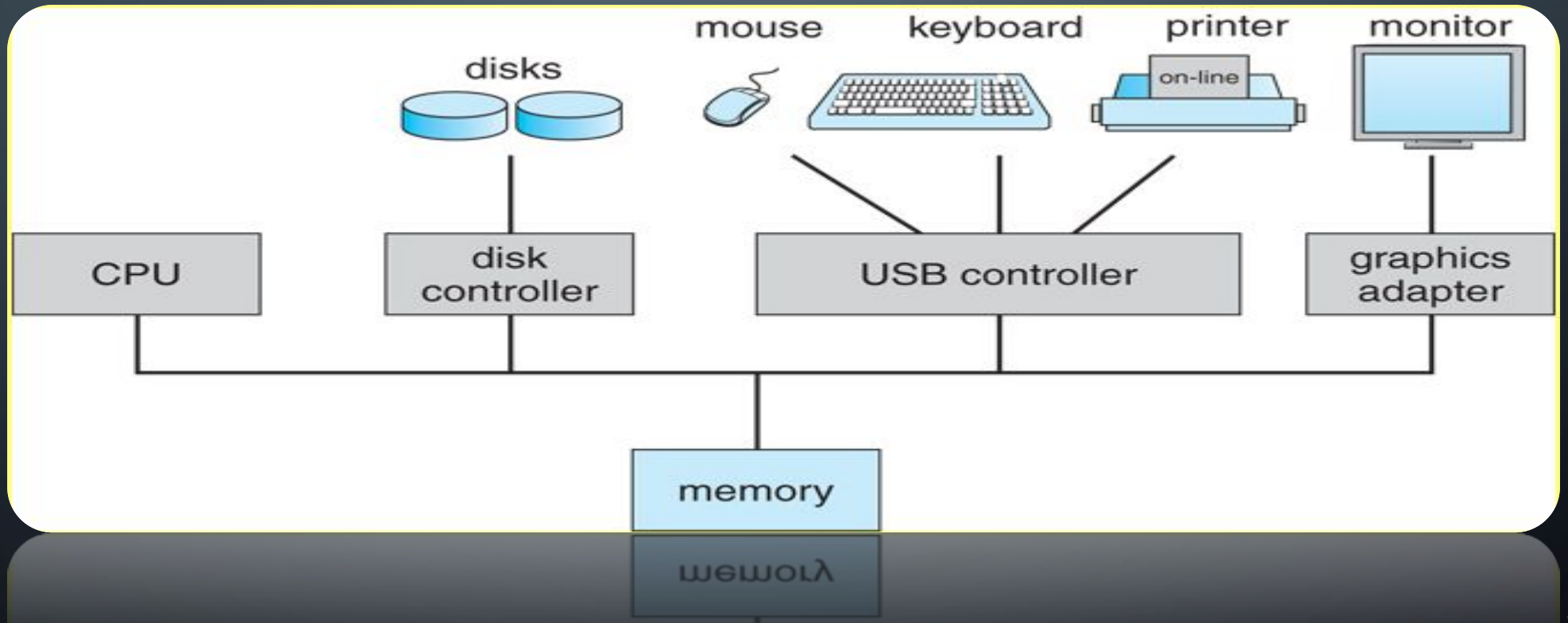
Computer systems consist of software and hardware that are combined to provide a tool to implement solutions for specific problems in an efficient manner and to execute programs.

COMPUTER SYSTEM ORGANIZATION

- Hardware
- Operating System
- Application Programs
- Users


user 1
user 2
user 3
...
user n
compiler
assembler
text editor
...
database system
system and application programs
operating system
computer hardware

COMPUTER SYSTEM ORGANIZATION



- The primary purpose of a computer system is to generate executable programs and execute them.
- The following are some of the main issues involved in performing these tasks.

1. Storing an executable on a secondary storage device such as hard disk
2. Loading executable from disk into the main memory
3. Setting the CPU state appropriately so that program execution could begin
4. Creating multiple cooperating processes, synchronizing their access to shared data, and allowing them to communicate with each other

WHAT IS AN OPERATING SYSTEM

- There are two views about this.
- The top-down view is that it is a program that acts as an intermediary between a user of a computer and the computer hardware and makes the computer system convenient to use.
- The bottom-up view is that it is a program, that allocates and deallocates computer system resources in an efficient fair, and secure manner- A resource manager
- A slightly different view of an OS emphasizes the need to control the various I/O devices and programs.

SINGLE-USER SYSTEMS

- Allows only one user to use the computer at a given time.
- The goals of such systems are maximizing user convenience and responsiveness, instead of maximizing the utilization of the CPU and peripheral devices.
- They can adopt technology developed for larger operating systems.
- Often individuals have sole use of computers and do not need advanced CPU utilization and hardware protection features.
- They may run different types of operating systems, including DOS, Windows, and MacOS. Linux and UNIX operating systems can also be run in single-user mode

BATCH SYSTEMS

- Early computers were large machines run from a console with card readers and tape drives as input devices and line printers, tape drives, and card punches as output devices.
- User did not interact directly with the system; instead the user prepared a job, which consisted of the program, data, and some control information about the nature of the job.
- The job was in the form of punch cards, and at some later time the output was generated by the system.
- Output=> result of the program + dump of the final memory and register contents for debugging

BATCH SYSTEMS

- To speed up processing, operators batched together jobs with similar needs and ran them through the computer as a group. For example, all FORTRAN programs were compiled one after the other.
- The major task of such an operating system was to transfer control automatically from one job to the next.
- The CPU is often idle because the speeds of the mechanical I/O devices such as a tape drive are slower than that of electronic devices.
- Digital Equipment Corporation's VMS is an example of a batch operating system.

MULTI-PROGRAMMED SYSTEMS

- Increases CPU utilization by organizing jobs so that the CPU always has one to execute.
- The operating system keeps several jobs in memory simultaneously.
- Since the number of jobs that can be kept simultaneously in memory is usually much smaller than the number of jobs that can be in the job pool.
- The operating system picks and executes one of the jobs in the memory.
- The job has to wait for some task such as an I/O operation to complete.
- In a non-multi-programmed system, the CPU would sit idle.
- In a multi-programmed system, the operating system simply switches to, and executes another job.

TIME-SHARING SYSTEMS

- Multi-user, multi-process, and interactive system.
- Allows multiple users to use the computer simultaneously.
- A user can run one or more processes at the same time and interact with his/her processes.
- A time-shared system uses multiprogramming and CPU scheduling to provide each user with a small portion of a time-shared computer.
- Each user has at least one separate program in memory.
- To obtain a reasonable response time, jobs may have to be swapped in and out of the main memory.
- UNIX, Linux, Windows NT Server, and Windows 2000 server are timesharing systems.

REAL TIME SYSTEMS

- A real-time system has well-defined, fixed time constraints, and if the system does not produce output for input within the time constraints, the system will fail.
- Real time systems come in two flavors: **hard** and **soft**.
- A hard real-time system guarantees that critical tasks be completed on time.
- A less restrictive type of real-time system is a soft real-time system, where a critical real-time task gets priority over other tasks, and retains that priority until it completes.
- Example: Systems that control scientific experiments, medical imaging systems, industrial control systems, and certain display systems.

HARD REAL-TIME SYSTEM

- Requires that all delays in the system be completed on time.
- This goal requires that all delays in the system be bounded.
- Secondary storage of any sort is usually limited or missing, with data instead of being stored in short-term memory or in read-only memory.
- Most advanced operating system features are absent too.

SOFT REAL-TIME SYSTEM

- Where a critical real-time task gets priority over other tasks, and retains that priority until it completes.
- As in hard real-time systems, the operating system kernel delays need to be bounded.
- Soft real-time is an achievable goal that can be mixed with other types of systems,
- Whereas hard real-time systems conflict with the operation of other systems such as time-sharing systems, the two cannot be mixed.