Functions of an Operating System

The operating system (OS) is the core of the system software. It manages computer resources (memory, input/output devices) and provides an interface for human-computer interaction (HCI).

Computer hardware is wired to initially load a small set of system instructions stored in permanent (nonvolatile) memory (ROM). Its popular name, BIOS, stands for Basic Input/Output System. BIOS boots the computer by loading a larger portion of systems software, usually from the hard disk. Nowadays, BIOS usually resides on EEPROM (Electrically Erasable Programmable Read-Only Memory) or flash memory.

The terms dual-boot and multi-boot system apply to computers that have two or more operating systems, respectively.
Functions of an Operating System (cont’d)

*Multiprogramming* is the technique of keeping multiple programs in main memory at the same time.

*Memory management* means keeping track of what programs are in memory and where in memory they reside.

A program in execution is called a *process*. A process may get interrupted during execution. A *context switch* is the procedure of storing and restoring the state (*context*) of a CPU so that multiple processes can share a single CPU resource.

*Process management* means keeping track of information for active processes.

*CPU scheduling* determines which process in memory is executed by the CPU at any given point.
Batch Processing

In a modern operating system, a batch is a system in which programs and system resources are coordinated and executed without interaction between the user and the programs.
Time-Sharing

*Time-sharing* refers to sharing a computing resource among many users by *multitasking*. Multitasking is a method by which multiple tasks, also known as processes, share common processing resources such as a CPU.

In a time-sharing system, each user has a *virtual machine*.

Early time-sharing systems were written for large *mainframes*.

Modern time-sharing systems support connection of multiple users to computers via networks.
Real-Time Systems

A real-time system is a system in which response time is crucial given the nature of the application domain.
Response Time

*Response time* is the time between receiving a stimulus and producing a response.
Memory Management

A *logical address* is a reference to a stored value relative to the program making the reference.

A *physical address* is an actual address in the main memory device.

The mapping of a logical address to a physical address is called *address binding*.
Single Contiguous Memory Management

Operating system

Application program

physical_address = base + logical_address
Partition Memory Management

- *Fixed-partition technique*: Main memory is divided into a specific number of partitions into which programs are loaded.
- *Dynamic-partition technique*: Main memory is divided into partitions as needed to accommodate programs.
- The *base register* holds the beginning address of the current partition.
- The *bounds register* stores the length of the current partition.
- Three general approaches to partition selection:
  - *First fit*: The first partition big enough to hold the program is allocated to it.
  - *Best fit*: The smallest partition big enough to hold the program is allocated to it.
  - *Worst fit*: The largest partition big enough to hold the program is allocated to it.
Paged Memory Management

Processes are divided into fixed-size pages and stored in memory frames when loaded.

The operating system maintains a separate page-map table (PMT) for each process in memory.

A logical address is invalid if (a) the page number is out of bounds for the process, and/or (b) the offset is larger than the frame size.

**Demand paging:** Pages are brought into memory only when referenced (on demand).

**Virtual memory:** The illusion that there is no restriction on program size.

**Thrashing:** Inefficient processing caused by constant page swapping.
Process Management

- Process states: new [being created], ready [waiting for the CPU only], waiting [for resources other than the CPU], running, terminated.

- Process control block (PCB) is the data structure used by the operating system to manage information about a process.
CPU Scheduling

- **Non-preemptive scheduling**: The currently executing process must give up the CPU voluntarily in order for another process to run.
- **Preemptive scheduling**: The operating system may decide to favor another process, preempting the current process.
- **Turnaround time**: The time elapsed between a process’s arrival in the ready state and its termination.
Scheduling Approaches

- *First-Come, First-Served (FCFS)*: Non-preemptive
- *Shortest Job Next (SJN)*: Also non-preemptive; it’s hard to know which job would run for the shortest time!
- *Round robin*: A *time slice* given to each process before being preempted is established