

Chapter 7

Operating Systems

OBJECTIVES

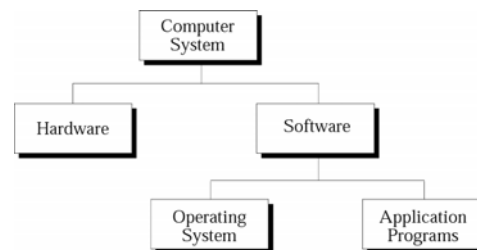
- Define the purpose and functions of an operating system.
- Understand the components of an operating system.
- Understand the concept of virtual memory.
- Understand the concept of deadlock and starvation.
- List some of the characteristics of popular operating systems such as Windows XP, Windows 2000, UNIX, Linux, MacOS.

Reading

- Chapter 7 (skip material related to deadlock and starvation on pages 132-135)

Figure 7-1

Computer System

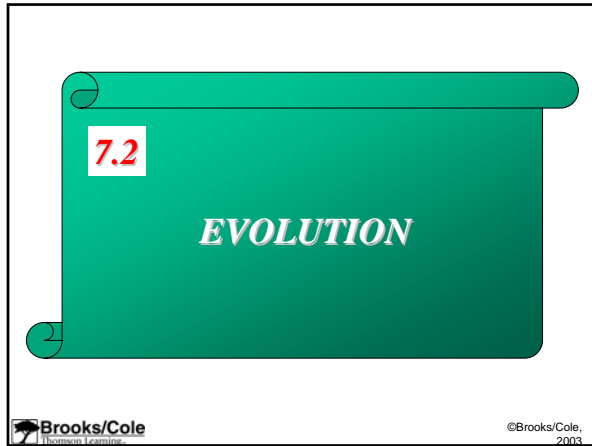


7.1

DEFINITION

Note:

An operating system is an interface between the hardware of a computer and the user (program or human) that facilitates the execution of the other programs and the access to hardware and software resources.

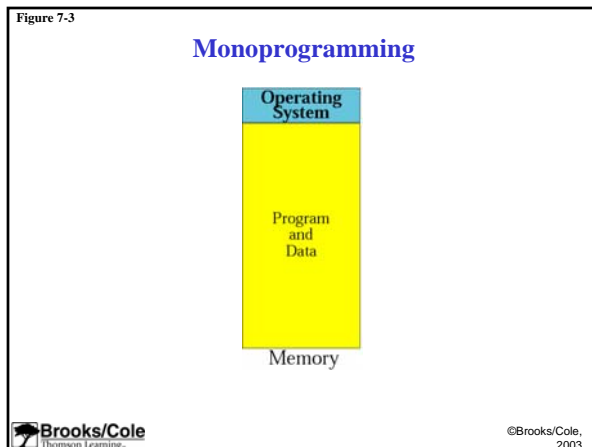
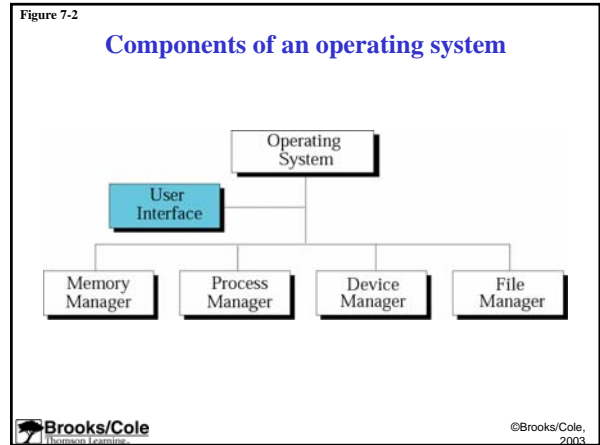
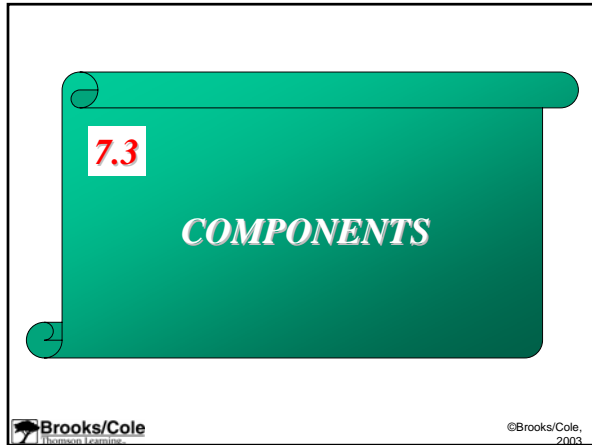


Evolution

- Batch Systems (OS is simple)
- Time-Sharing Systems (multiprogramming, time-sharing)
- Personal Systems (single-user operating systems)
- Parallel Systems (multiple CPUs on the same machine)
- Distributed Systems (remote sharing of resources)

Brooks/Cole
Thomson Learning

©Brooks/Cole, 2003



Multiprogramming

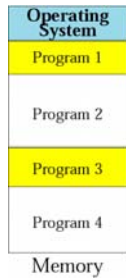
- More than one program is in memory at the same time
- They are executed concurrently
- The CPU switches between the programs

Brooks/Cole
Thomson Learning

©Brooks/Cole, 2003

Figure 7-4

Multiprogramming



Multiprogramming

- Swapping means that during execution the program can be swapped between memory and disk one or more times

Figure 7-5

Categories of multiprogramming

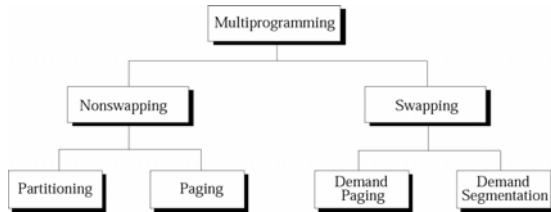
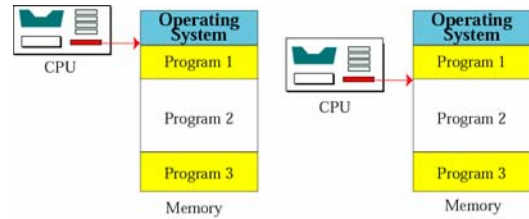


Figure 7-6

Partitioning



a. CPU starts executing program 1. b. CPU starts executing program 2.

Figure 7-7

Paging

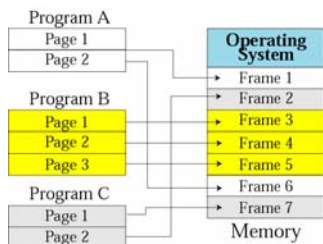
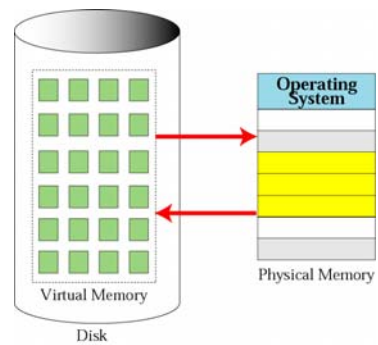


Figure 7-8

Virtual memory

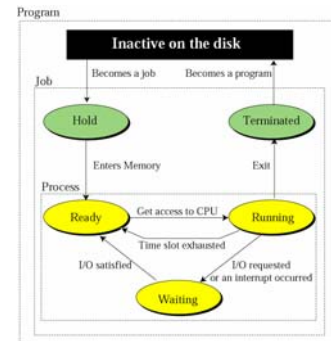


Process Management

- **Program** is a non active set of instructions written by a programmer and stored on disk
- **Job** is a program that becomes a job from the moment it is selected for execution until it has finished running and becomes a program again
- **Process** is a program in execution (a job that is residing in memory)

Figure 7-9

State diagram with the boundaries between a program, a job, and a process



Process Management

Process manager uses two schedulers:

- The job scheduler
- The process scheduler

Figure 7-10

Job scheduler

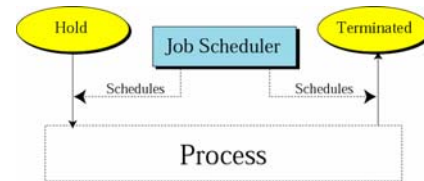
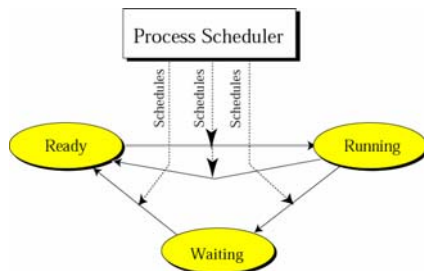


Figure 7-11

Process scheduler

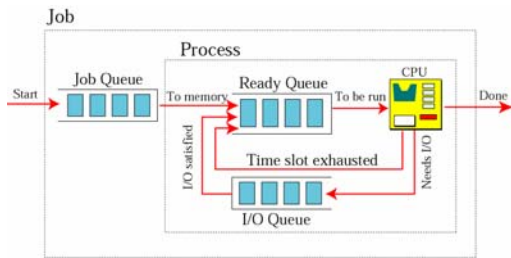


Queuing

- To handle multiple processes and jobs, the process manager uses queues (waiting lists)
- An operating system can have several queues
- The process manager can have different policies for selecting the next job or process from a queue: FIFO, shortest length first, highest priority

Figure 7-12

Queues for process management



Synchronization

- The main idea of a process management is to synchronize different processes with different resources
- You can have two possible situations whenever resources are used by more than one user: deadlock and starvation

Figure 7-13

Deadlock

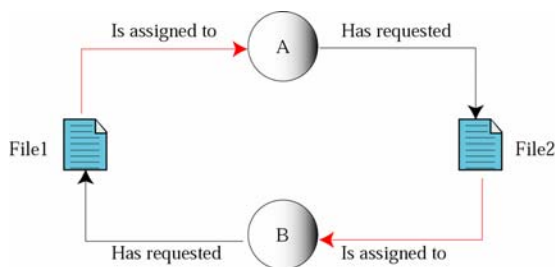
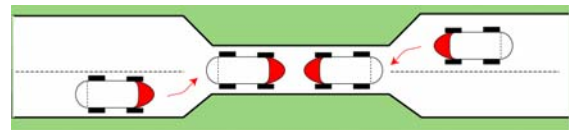


Figure 7-14

Deadlock on a bridge

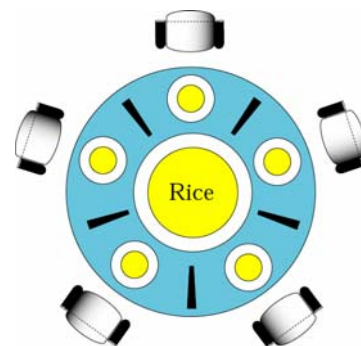


Note:

Deadlock occurs when the operating system does not put resource restrictions on processes.

Figure 7-16

Dining philosophers



Device Manager

- Monitors every input/output device to assure that the device is functioning properly
- Maintains a queue for each input/output device
- Controls different policies for accessing I/O devices

File Manager

- Controls access to files
- Supervises the creation, deletion and modification of files
- Can control the naming of files
- Supervises the storage of files
- Responsible for archiving and backups

User Interface

- User interface is a program that accepts requests from users and interprets them for the rest of the operating system
- UI in UNIX is called a **shell**
- UI in Windows is comprised of GUI (graphical user interface)

7.4

POPULAR OPERATING SYSTEMS

Popular Operating Systems

- Windows XP
- Windows 2000
- Windows 95/98/NT
- UNIX
- Linux
- MacOS

Summary

- An operating system facilitates the execution of other software, acts as the general manager of a computer system and ensures the efficient use of hardware and software resources
- The evolution of OS has included batch operating systems, time-sharing, single-user, parallel and distributed systems
- The OS has the memory manager, the process manager, device manager and the user interface
- In monoprogramming, most of memory capacity is dedicated to one single program
- In multiprogramming, more than one program is in memory at the same time

Summary

- In partitioning, memory is divided into variable length sections
- In paging, memory is divided into equally sized sections called frames and the program is divided into equally sized section called pages
- The sum of the sizes of all the programs in memory is virtual memory
- A program is a nonactive set of instruction written by a programmer and stored on disk
- A job is a program that is selected for execution
- A process is a job residing in memory

Summary

- A state diagram shows the relationship between a program, job and process
- The job scheduler creates a process from a job and changes a process back to a job
- The process scheduler moves a process from one state to another
- Jobs and processes wait in queues
- The device manager controls access to I/O devices
- The file manager controls access to files
- The user interface is software that accepts requests from processes and interprets them for the rest of OS
- Popular OS: Windows XP, Unix, Linux, MacOS