Operating System

Lecture 05

Chap 10 (In OS book, uploaded to sandata)

Software Categories

Application software

Software written to address specific needs —to solve problems in the real world

System software

Software that manages a computer system at a fundamental level

Can you name examples of each?

Roles of an Operating System

Operating system

System software that

- **manages** computer resources, such as memory and input/output devices
- **provides** an interface through which a human can interact with the computer
- **allows** an application program to interact with these other system resources

Roles of an Operating System

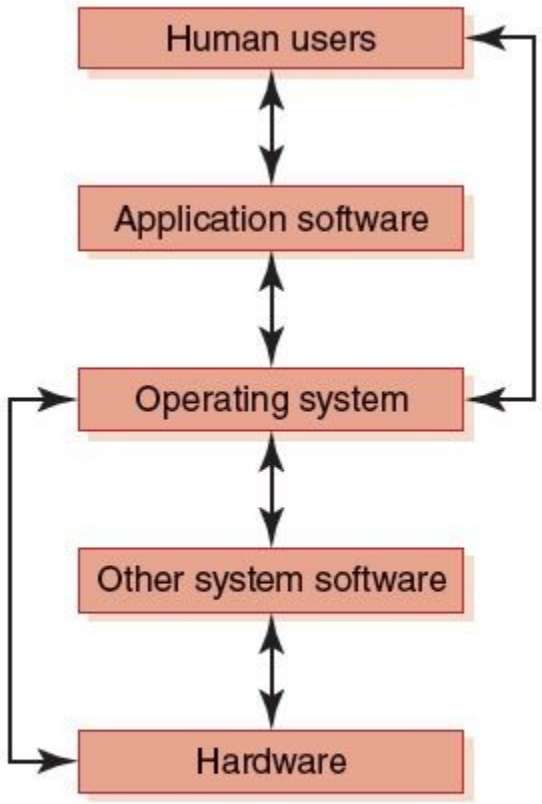


FIGURE 10.1 An operating system interacts with many aspects of a computer system.

What operating systems have you used?

Roles of an Operating System

Booting

Hardware is wired to initially load a small set of system instructions stored in ROM

These instructions load a larger set of instructions from hard disk

Dual boot - Multiboot

Roles of an Operating System

The various roles of an operating system generally revolve around the idea of "sharing nicely"

An operating system manages resources, and these resources are often shared in one way or another among programs that want to use them

Resource Management

Multiprogramming

The technique of keeping multiple programs that compete for access to the CPU in main memory at the same time so that they can execute

Memory management

The process of keeping track of what programs are in memory and where in memory they reside

Resource Management

Process

A program in execution

Process management

The act of carefully tracking the progress of a process and all of its intermediate states

CPU scheduling

Determining which process in memory is executed by the CPU at any given point

Memory Management

Memory Management (MM)

We will study the following MM algorithms

1. Single Contiguous Memory Management
2. Partition Memory Management

Memory Management

Operating systems must employ techniques to

- Track where and how a program resides in memory
- Convert **logical addresses** into actual **addresses**

Logical address

Reference to a stored value relative to the program making the reference

Physical address

Actual address in main memory

1. Single Contiguous MM

FIGURE 10.5 Main memory divided into two sections

There are only two programs in memory

- The operating system
- The application program

This approach is called **single contiguous memory management**

Continue…

In concrete terms:

A logical address is simply an integer value relative to the starting point of the program

A physical address is a logical address added to the starting location of the program in main memory

Advantages – Disadvantages ?

Continue…

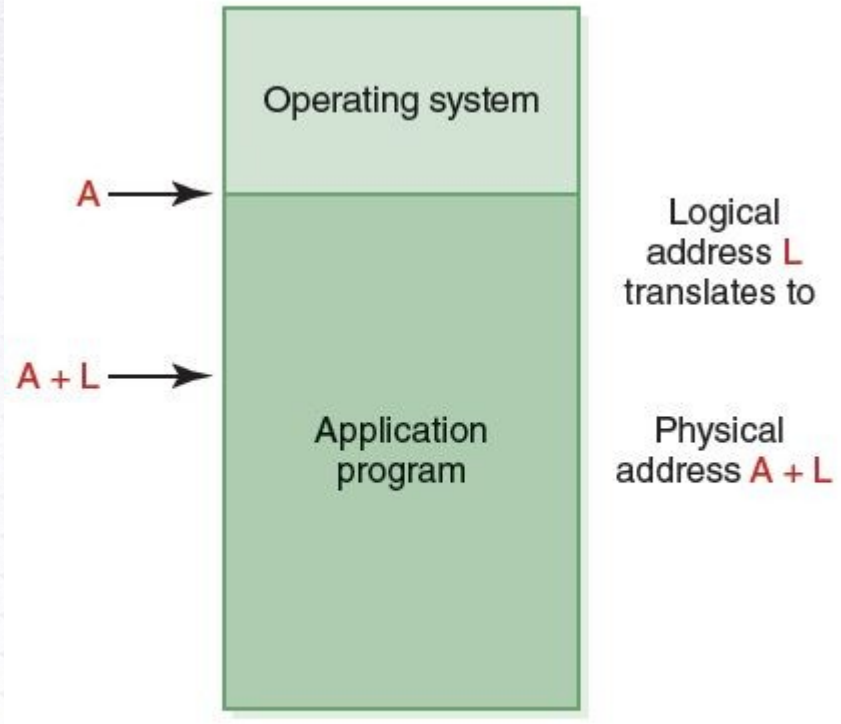


FIGURE 10.6 Binding a logical address to a physical address

If A is location 100, and the application program is Program 1, then sum is stored at location 123.

2. Partition Memory Management

Single contiguous MM has only the OS and one other program in memory at one time

Partition MM has the OS and any number of other programs in memory at one time

There are two schemes for dividing up memory for programs:

- **Fixed partitions** Main memory is divided into a fixed number of partitions into which programs can be loaded
- **Dynamic partitions** Partitions are created as needed to fit the programs waiting to be loaded

Continue…

Memory is divided into a set of partitions, some empty and some allocated to programs

Base register

A register that holds the beginning address of the current partition (the one that is running)

Bounds register

A register that holds the length of the current partition

Continue…

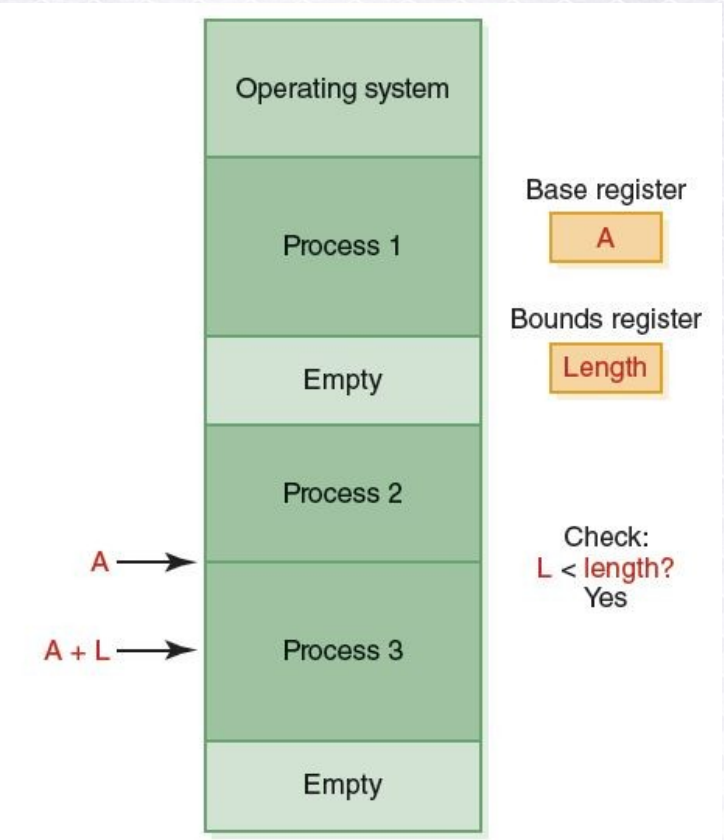


FIGURE 10.7 Address resolution in partition memory management

Why check?

Continue…

Which partition should we allocate to a new program?

- **First fit** Allocate program to the first partition big enough to hold it
- **Best fit** Allocate program to the smallest partition big enough to hold it
- **Worst fit** Allocate program to the largest partition big enough to hold it

Continue…

A: 1000
B: 700
C: 750
D: 1500
E: 300
F: 350

Requests come in for blocks of the following sizes:
1000, 25, 780, 1600, and 325

What block will be assigned to each request if the

- *first-fit algorithm is used?*
- *best-fit algorithm is used?*
- *worst-fit algorithm is used?*

(Treat each request as an independent event)

Process Management

Process Management

Process management

The act of managing the use of the CPU by individual processes

Recall that a process is a program in execution

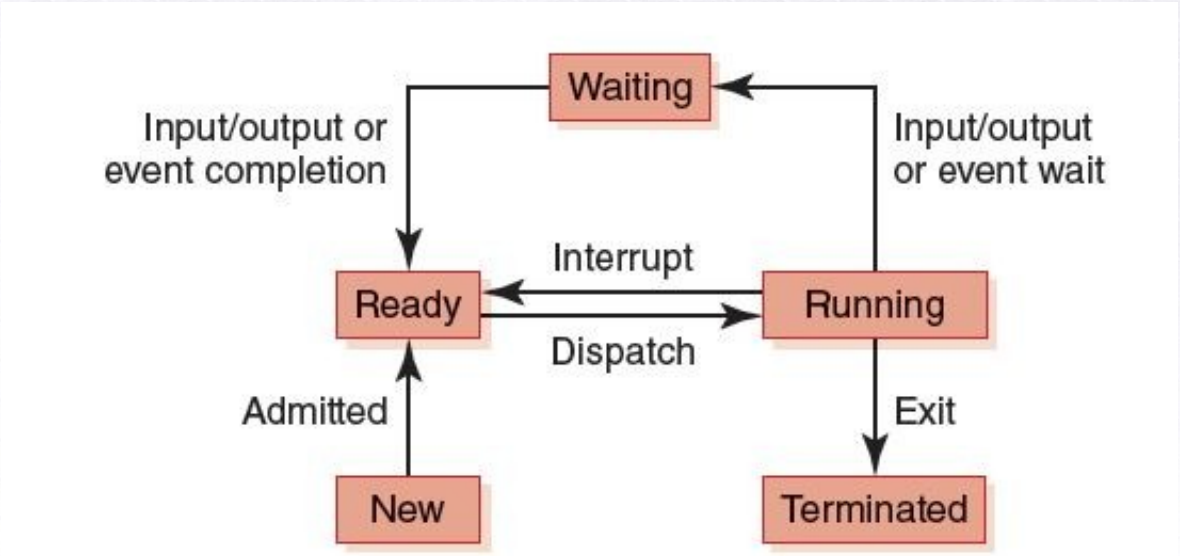
What stages does a process go through?

Process Management
(cont…)

- To understand how an operating system manages processes, we must recognize the stages that a process goes through during its computational life
- And understand the information that must be managed to keep a process working correctly in a computer system.

Process Management

The Process States



What can cause a process to move to The Waiting state?

Process Management

Process control block (PCB)

A *data structure* used by the OS to manage information about a process, including

- current value of the program counter
- values of all CPU registers for the process
- base and bound register values (or page tables)
- accounting information

Each *state* is represented by a list of PCBs, one for each process in that state

Process Management

There is only one CPU and therefore only one set of CPU registers, which contain the values for the currently executing process

Each time a process is moved to the running state:

- Register values for the currently running process are stored into its PCB
- Its PCB is moved to the list of the state into which it goes
- Register values of the new process moving into the running state are loaded into the CPU
- This exchange of register information is called a **context switch**

CPU Scheduling

CPU Scheduling

CPU Scheduling

The act of determining which process in the *ready* state should be moved to the *running* state

- Many processes may be in the ready state
- Only one process can be in the running state, making progress at any one time

Which one gets to move from ready to running?

CPU Scheduling

Non-preemptive scheduling

The currently executing process gives up the CPU voluntarily

Preemptive scheduling

The operating system decides to favor another process, preempting the currently executing process

Turnaround time

- An evaluation method of scheduling algorithms
- The amount of time between when a process arrives in the **ready state** the first time and when it exits the **running state** for the last time

CPU Scheduling Algorithms

First-Come, First-Served

Processes are moved to the CPU in the order in which they arrive in the running state

Shortest Job Next

Process with shortest estimated running time in the ready state is moved into the running state first

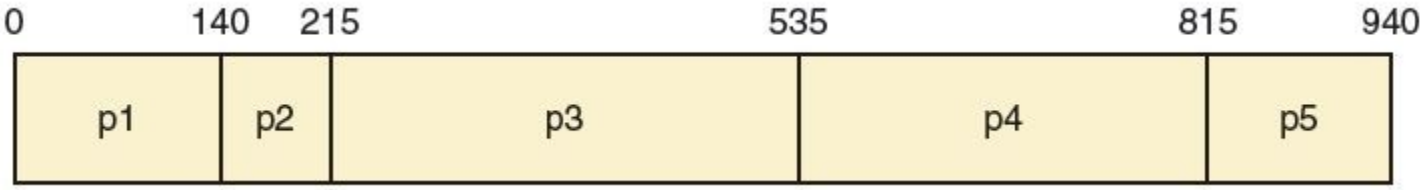
Round Robin

Each process runs for a specified time slice and moves from the running state to the ready state to await its next turn if not finished

First-Come, First-Served

Process	Service Time
p1	140
p2	75
p3	320
p4	280
p5	125

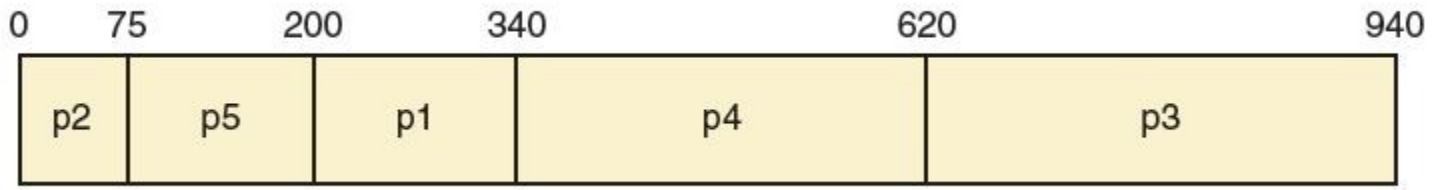
What is the average turn-around time?



Shortest Job Next

Process	Service Time
p1	140
p2	75
p3	320
p4	280
p5	125

What is the average turn-around time?



Round Robin

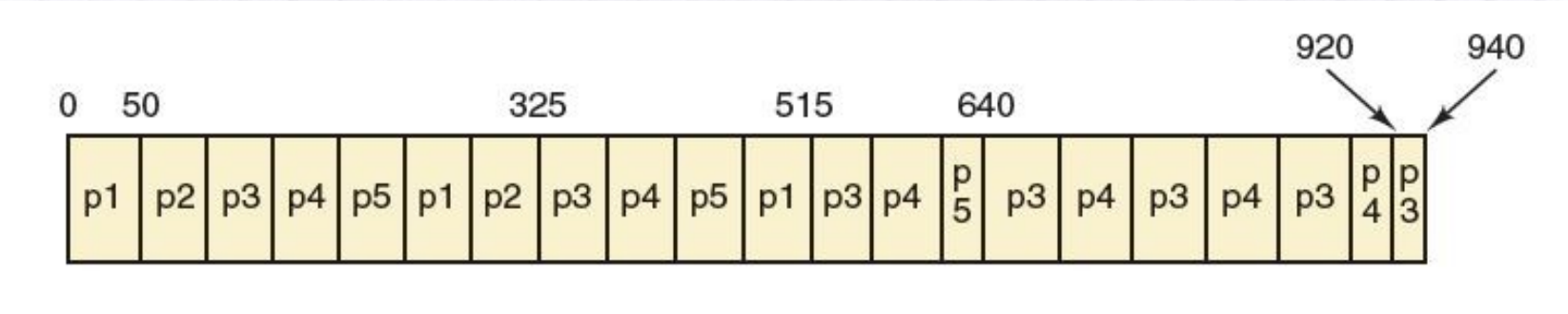
Every process is treated the same!

Time slice (quantum)

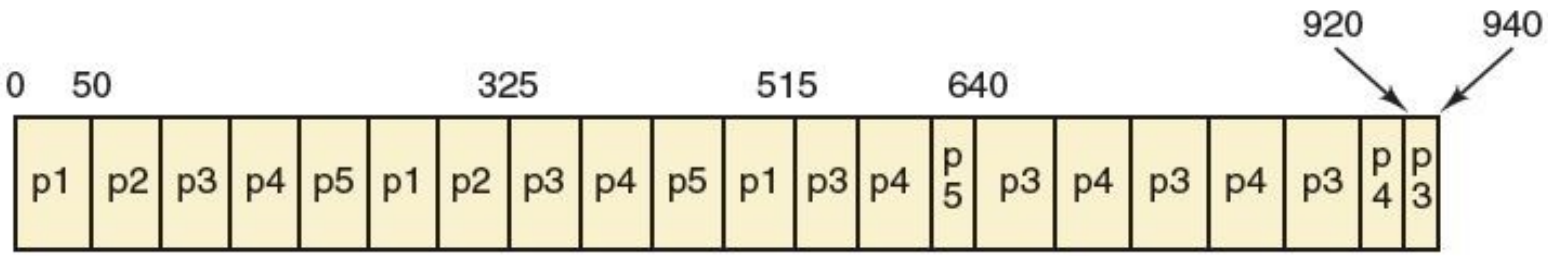
The amount of time each process receives before being preempted and returned to the ready state to allow another process its turn

Round Robin

Suppose the time slice is 50



What is the average turnaround time?



The average turnaround time for this example is (515 + 325 + 940 + 920+ 640) / 5, or 668.

Note that this turnaround time is higher than in the other examples. Does that mean the round-robin algorithm is not as good as the others?

No

CPU Scheduling Algorithms

Are these scheduling algorithms preemptive or non-preemptive? Explain

First-Come, First-Served?

Shortest Job Next?

Round Robin?

Activity 1:

Use the following table of processes and service time. Draw Gantt charts that show turnaround time for each process using SJN (shortest job next) and RR (round robin) CPU Scheduling. Also find average turnaround time for each.

Process	P1	P2	P3	P4	P5
Service time	120	60	180	50	300

Activity 2:

Apply First-Fit and Best-Fit Algorithm on the following jobs given along the memory set available (fixed partitions). Fill Job Number column for your answer.

MEMORY MAP

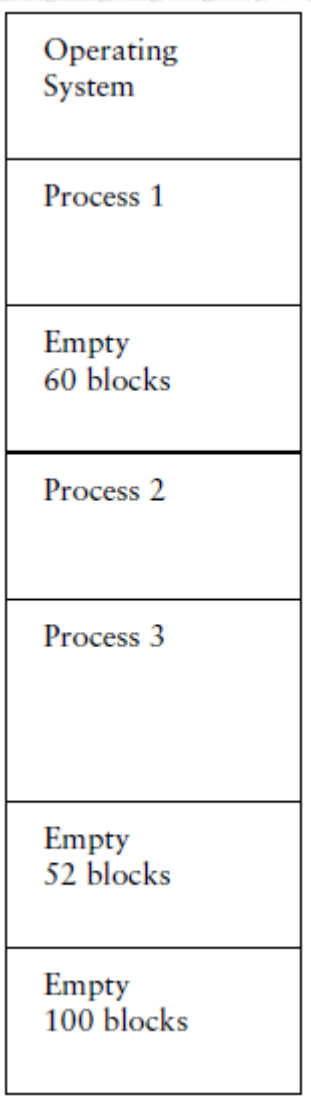
	First-Fit (Job No)	Best-Fit (Job No)
8 KB		
100 KB		
1024 KB		
56 KB		
2 KB		
48 KB		
2048 KB		
4 KB		
8 KB		
12 KB		
56 KB		

Jobs

J1	J2	J3	J4	J5	J6	J7	J8	J9	J10
4 KB	30 KB	6 KB	9 KB	10 KB	100 KB	200 KB	2 KB	5 KB	1 KB

Activity 3:

- If the partitions are fixed and a new job arrives requiring 52 blocks of main memory, show memory after using each of the following partition selection approaches:
- **a.** first fit
- **b.** best fit
- **c.** worst fit



Activity 4:

- If the partitions are dynamic and a new job arrives requiring 52 blocks of main memory, show memory after using each of the following partition selection approaches:
- **a.** first fit
- **b.** best fit
- **c.** worst fit