# Chapter 2

# Memory Management: Simple Systems

**A Guide to this Instructor's Manual:**

We have designed this Instructor's Manual to supplement and enhance your teaching experience through classroom activities and a cohesive chapter summary.

This document is organized chronologically, using the same headings that you see in the textbook. Under the headings you will find: lecture notes that summarize the section, Teacher Tips, Classroom Activities, and Lab Activities. Pay special attention to teaching tips and activities geared towards quizzing your students and enhancing their critical thinking skills.

In addition to this Instructor's Manual, our Instructor's Resources also contain PowerPoint Presentations, Test Banks, and other supplements to aid in your teaching experience.

**At a Glance**

## Instructor's Manual Table of Contents

Lecture Notes

# Overview

This chapter introduces the role of main memory (also known as random access memory or RAM, core memory, or primary storage) and four types of memory allocation schemes: single-user systems, fixed partitions, dynamic partitions, and relocatable dynamic partitions. The discussion begins with the simplest memory management scheme – the one used with the simplest computer systems.

# Learning Objectives

After completing this chapter, the student should be able to describe:

- The basic functionality of the four memory allocation schemes presented in this chapter: single user, fixed partitions, dynamic partitions, and relocatable dynamic partitions
- Best-fit memory allocation as well as first-fit memory allocation
- How a memory list keeps track of available memory
- The importance of memory deallocation
- The importance of the bounds register in memory allocation schemes
- The role of compaction and how it can improve memory allocation efficiency

# Teaching Tips

## Single-User Contiguous Scheme

1. With this memory allocation scheme, each job or program is loaded in its entirety into memory and allocated as much contiguous space in memory as it needs before execution can begin. Use Figure 2.1 to aid the discussion.

2. Note that this scheme demonstrates a significant limiting factor of all computers – they have only a finite amount of memory.

3. Discuss the steps involved in allocating memory.

4. Point out that one major problem with this type of memory allocation scheme is that it does not support multiprogramming (multiple jobs or processes occupying memory at the same time).

### Fixed Partitions

1. Introduce the term **fixed partitions**.

2. Note the important factor that was introduced with this scheme – the protection of the job's memory space.

3. Discuss the algorithm used to store jobs in memory. Note that it requires a few more steps than the one used for a single-user system because the size of the job must be matched with the size of the available partitions to make sure it fits completely.

4. In order to allocate memory spaces to jobs, the Memory Manager must maintain a table which shows each memory partition's size, its address, its access restrictions, and its current status (free or busy). Use Table 2.1 and Figure 2.2 to aid the discussion.

5. Point out that the fixed partition scheme works well if all of the jobs that run on the system are of similar size or if the sizes are known ahead of time and do not vary between reconfigurations.

6. Discuss the consequences if the partition sizes are too small or are too big. Use Figure 2.3 to aid the discussion.

7. Introduce the term **internal fragmentation**.

| | |
|---|---|
| ***Teaching Tip*** | Fixed partitioning allowed multiprogramming, but arbitrary partition sizes led to the problem of internal fragmentation. |

### Dynamic Partitions

1. With the introduction of the dynamic partition allocation scheme, memory is allocated to an incoming job in one contiguous block, and each job is given only as much memory as it requests when it is loaded for processing. Discuss the problems this introduces. Use Figures 2.3 and 2.4 to aid the discussion.

## Quick Quiz 1

1. Which of the following memory management schemes does not allow multiprogramming?
   a. Single-user contiguous scheme
   b. Fixed partition
   c. Dynamic partition
   d. Relocatable dynamic partitions
   Answer: a

2. Which of the following are the disadvantages of a fixed partition scheme (choose all that apply)?
   a. It requires that the entire program be loaded into memory.
   b. It requires that the entire program be stored contiguously.
   c. It requires that the entire program remain in memory until the job is completed.
   d. It does not allow multiprogramming.

Answer: a, b, and c

3. The phenomenon of less-than-complete use of memory space in a fixed partition is called ____.
   Answer: internal fragmentation

4. (True or False) There are two types of fragmentation: internal and external. The type depends on the location of the wasted space.
   Answer: True

## Best-Fit and First-Fit Allocation

1. Explain that for both fixed and dynamic memory allocation schemes, the operating system must keep lists of each memory location, noting which are free and which are busy.

2. Discuss the difference between the best-fit and first-fit allocation methods.

3. Students should understand the trade-offs between the two concepts. Use the example described on pages 36-38 and Figure 2.4 to aid the discussion

4. Use Figure 2.5 to show how a large job can have problems with a first-fit memory allocation list. Compare the results with that shown in Figure 2.6, which demonstrates that the same job list, using a best-fit scheme, would use memory more efficiently.

5. Discuss the first-fit algorithm (included in Appendix A), which assumes that the Memory Manager keeps two lists, one for free memory blocks and one for busy memory blocks. Use Table 2.2 to aid the discussion.

6. Introduce the best-fit algorithm, which is slightly more complex because the goal is to find the smallest memory block into which the job will fit. Use Figure 2.6 to aid the discussion.

7. Note that the best-fit algorithm (included in Appendix A) is illustrated showing only the list of free memory blocks. Table 2.3 shows the free list before and after the best-fit block has been allocated to the same request presented in Table 2.2.

| ***Teaching Tip*** | To learn more about the first-fit allocation scheme, visit: http://www.memorymanagement.org/articles/alloc.html |
|---|---|

## Deallocation

1. Introduce the concept of deallocation.

2. Note that a dynamic partition system uses a more complex algorithm (shown in Appendix A) because it tries to combine free areas of memory whenever possible. Therefore, the system must be prepared for three alternative situations:
   **Case 1: When the block to be deallocated is adjacent to another free block**
   **Case 2: When the block to be deallocated is between two free blocks**
   **Case 3: When the block to be deallocated is isolated from other free blocks**

**Case 1: Joining Two Free Blocks**

1. Table 2.4 shows how deallocation occurs in a dynamic memory allocation system when the job to be deallocated is next to one free memory block. Note that after deallocation, the free list looks like the one shown in Table 2.5.

**Case 2: Joining Three Free Blocks**

1. Use Table 2.6 and Table 2.7 to discuss the process involved in joining three free blocks.

2. Introduce the term **null entry**.

**Case 3: Deallocating an Isolated Block**

1. The third alternative occurs when the space to be deallocated is isolated from all other free areas. Use the example on page 43 and Tables 2.8 and 2.9 to aid the discussion.

2. Note that the scheme presented in the example creates null entries in both the busy and the free lists during the process of allocation or deallocation of memory. Use Tables 2.10 and 2.11 to aid the discussion.

## Relocatable Dynamic Partitions

1. With this memory allocation scheme, the Memory Manager relocates programs to gather together all of the empty blocks in order to compact them to make one block of memory large enough to accommodate some or all of the jobs waiting to get in.

2. Introduce the term **compaction of memory**.

3. Note that compaction is not an easy task. Most or all programs in memory must be relocated so that they are contiguous, and then every address, and every reference to an address, within each program must be adjusted to account for the program's new location in memory.

4. Use the program shown in Figures 2.7 and 2.8 to illustrate how the operating system flags addresses so that they can be adjusted if and when a program is relocated. Figure 2.9 illustrates what happens to a program in memory during compaction and relocation.

5. Point out that the discussion of compaction raises three questions:
   - What goes on behind the scenes when relocation and compaction take place?

- What keeps track of how far each job has moved from its original storage area?
- What lists have to be updated?

6. Introduce the terms **bounds register** and **relocation register**. Use Figures 2.10 and 2.11 to aid the discussion.

7. Discuss the contents of the relocation register.

8. In effect, by compacting and relocating, the Memory Manager optimizes the use of memory and thus improves throughput - an important measure of system performance. An unfortunate side effect is that this memory allocation scheme requires more overhead than with the previous schemes. The crucial factor here is the timing of the compaction - when and how often it should be done. Discuss the three options available.

| ***Teaching Tip*** | To learn more about memory allocation schemes, visit: http://www.scribd.com/doc/59021292/Types-of-Memory-Allocation-Schemes |
|---|---|

## Quick Quiz 2

1. The ____ is used to store the highest location in memory accessible by each program.
   Answer: bounds register

2. The ____ contains the value that must be added to each address referenced in the program so that the system will be able to access the correct memory addresses after relocation.
   Answer: relocation register

3. The ____ is performed by the operating system to reclaim fragmented space.
   Answer: compaction of memory

4. (True or False) The operating system must distinguish between addresses and data values, and these distinctions are not obvious after the program has been loaded into memory.
   Answer: True

## Class Discussion Topics

1. Discuss the limitations of memory management schemes in early systems. What were the most problematic aspects of these schemes? Why were they sufficient for the first three generations of computers?

2. Discuss the compaction process. How often should it occur and what are its advantages and disadvantages?

## Additional Projects

1. Submit a report that discusses the differences between the next-fit and worst-fit algorithms.

2. List the steps involved in program relocation, including loop sequences, decision sequences, branching sequences, and data references.

## Additional Resources

1. Beginner's Guide to Memory Management: www.memorymanagement.org/articles/begin.html

2. Windows Memory Management: http://msdn.microsoft.com/en-us/library/aa366779%28VS.85%29.aspx

3. IBM: Inside Memory Management: www.ibm.com/developerworks/linux/library/l-memory/

5. Ravenbrook – The Memory Management Reference: www.memorymanagement.org

## Key Terms

- **address:** a number that designates a particular memory location.
- **best-fit memory allocation:** a main memory allocation scheme that considers all free blocks and selects for allocation the one that will result in the least amount of wasted space.
- **bounds register:** a register used to store the highest location in memory legally accessible by each program.
- **compaction of memory:** the process of collecting fragments of available memory space into contiguous blocks by relocating programs and data in a computer's memory.
- **deallocation:** the process of freeing an allocated resource, whether memory space, a device, a file, or a CPU.
- **dynamic partitions:** a memory allocation scheme in which jobs are given as much memory as they request when they are loaded for processing, thus creating their own partitions in main memory.
- **external fragmentation:** a situation in which the dynamic allocation of memory creates unusable fragments of free memory between blocks of busy, or allocated, memory.
- **first-fit memory allocation:** a main memory allocation scheme that searches from the beginning of the free block list and selects for allocation the first block of memory large enough to fulfill the request.
- **fixed partitions:** a memory allocation scheme in which main memory is sectioned with one partition assigned to each job.

- **internal fragmentation:** a situation in which a partition is only partially used by the program; the remaining space within the partition is unavailable to any other job and is therefore wasted.
- **main memory:** the unit that works directly with the CPU and in which the data and instructions must reside in order to be processed. Also called *random access memory (RAM), primary storage,* or *internal memory*.
- **null entry:** an empty entry in a list.
- **RAM:** another term for *main memory*.
- **relocatable dynamic partitions:** a memory allocation scheme in which the system relocates programs in memory to gather together all empty blocks and compact them to make one block of memory that's large enough to accommodate some or all of the jobs waiting for memory.
- **relocation register:** a register that contains the value that must be added to each address referenced in the program so that the Memory Manager will be able to access the correct memory addresses.
- **relocation:** (1) the process of moving a program from one area of memory to another; or (2) the process of adjusting address references in a program, by either software or hardware means, to allow the program to execute correctly when loaded in different sections of memory.