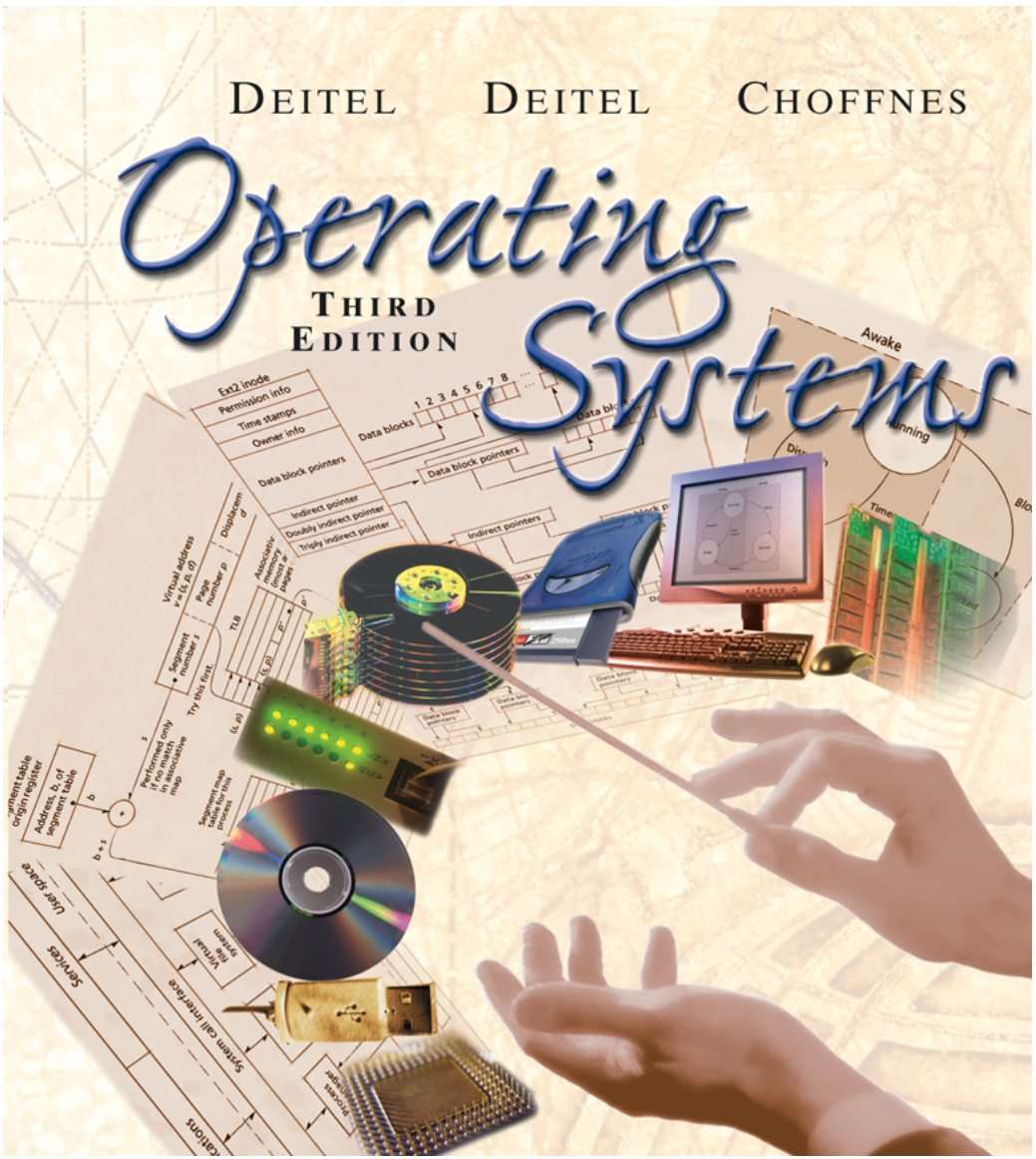
INSTRUCTOR'S MANUAL
FOR
OPERATING SYSTEMS, THIRD EDITION



DEITEL DEITEL CHOFFNES

Contents

Chapter 1

Introduction to Operating Systems

Chapter 1

Exercise Solutions

1.1 Distinguish between multiprogramming and multiprocessing. What were the key motivations for the development of each?

Ans: A multiprogramming system runs more than one program "simultaneously" on one processor. The system attempts to keep several programs resident in main memory and switches the processor rapidly between them. Multiprogramming was developed to improve processor and I/O resource utilization. A multiprocessor is a computer system with more than one processor. Multiprocessing was developed in an effort to increase processing speeds by allowing truly parallel computation.

1.2 Briefly discuss the significance of each of the following systems mentioned in this chapter:

a. MS-DOS

Ans: This is the world's first truly widely used personal computer operating system. It was developed by Microsoft for the original line of IBM Personal Computers.

b. CTSS

Ans: This is the Compatible Time-Sharing System developed at MIT in the early 1960s as one of the first effective timesharing operating systems.

c. Multics

Ans: Multics was the successor to CTSS at MIT. Multics evolved CTSS into a virtual memory operating system. In addition, Multics was the first major operating system written primarily in a high-level language.

d. OS/360

Ans: This was the operating system developed by IBM for its highly successful System/360 family of computers.

e. TSS

Ans: TSS was the timesharing operating system developed by IBM (for its System 360/67 virtual memory computer). Although TSS never became a successful commercial product, it gave IBM developers valuable experience in investigating virtual memory operating systems. This expertise proved helpful in the development of IBM's highly successful VM operating system.

f. UNIX

Ans: UNIX is more precisely referred to as UNIX systems. It is not a single operating system, but rather a whole family of operating systems that have evolved from the original system created at Bell Laboratories in the late 1960s.

g. Macintosh

Ans: The Apple Macintosh personal computer and its associated operating system popularized the notion of the user-friendly, mouse-oriented, graphical user interface (GUI) originally developed at Xerox's Palo Alto Research Laboratories.

1.3 What developments made personal computing feasible?

Ans: The key developments were the availability of the microprocessor and economical memories.

1.4 Why is it impractical to use a virtual machine for a hard real-time system?

Ans: Hard real-time systems must perform certain tasks immediately. This requires that the system has immediate access to hardware when those tasks must be performed. Since a virtual machine is just an extra software layer between the application and the hardware, using a virtual machine would be inefficient.

1.5 What role did the development of graphical user interfaces play in the personal computer revolution?

Ans: GUIs made computers easier to use (i.e., "user friendly") by enabling users to access computer services quickly and conveniently using a mouse pointer and icons.

1.6 The GNU Public License (GPL) promotes software that is free, as in "freedom." How does the GPL provide such freedom?

Ans: The GPL promotes freedom by enabling any software under its license to be freely shared and modified. This software can be integrated into commercial products as long as they are distributed under the GPL.

1.7 How has distributed computing affected operating system design?

Ans: With distributed computing, operating systems manage resources outside a single computer. For example, data could be accessed from, and processes could be run on, remote computers. This required that operating systems designers become familiar with networking issues, which we discuss in Chapter 16, Introduction to Networking.

1.8 What are the advantages and disadvantages of communication between computers?

Ans: Communication between computers improves productivity by enabling users to share information and improves efficiency and computing power by enabling multiple computers to share resources. However, communication enables a third party to intercept transmitted data, a security risk that encryption attempts to minimize. It also occurs at relatively slow speeds compared to the internal processing speeds of individual computers. This affects the kinds of applications that can be implemented involving intercomputer communication.

1.9 Define, compare, and contrast each of the following terms:

a. online

Ans: This simply means that a computer is turned on and directly connected to the network. For example, when their PCs are on, hundreds of millions of users are online because their PCs are connected to the Internet.

b. real time

Ans: Real-time systems attempt to respond to a request within a bounded amount of time. These types of systems are useful for applications that must react rapidly to changes in an environment. A hard real-time system meets its goals only if all tasks complete on time. A soft real-time system performs its real-time tasks with higher priority but does not ensure that they will always complete on time.

c. interactive computing

Ans: Interactive computing involves a user being present when jobs or processes are run and carrying on a dialog with the machine. Interactive computing ordinarily involves a series of short user requests that require prompt service.

d. timesharing

Ans: Timesharing is essentially multiprogramming of multiple interactive users.

1.10 How do middleware and Web services promote interoperability?

Ans: Middleware links two separate applications (e.g., a Web server and a database), simplifying communication across different architectures. Examples of middleware include the CORBA, RMI and DCOM interfaces, which are discussed in Chapter 17, Introduction to Distributed Systems. Web services define standards that can enable any two computer applications to communicate and exchange data via the Internet.

1.11 Evaluate monolithic, layered and microkernel architectures according to

a. efficiency.

b. robustness.

c. extensibility.

d. security.

Ans: The primary advantage to monolithic kernels is that they are efficient. Because all operating system components are contained in the kernel, developing a robust, extensible and secure monolithic operating system can be challenging. Layered architectures are more extensible, easier to debug and potentially easier to secure than monolithic architectures because they are more modular. However, layering reduces efficiency. Microkernel operating systems are highly extensible, because many operating system components execute independently of one another. This increased modularity can also enhance a microkernel architecture's robustness and security. This high level of modularity, however, makes microkernels less efficient than other operating system architectures due to interprocess communication overhead.

Suggested Projects

1.12 Prepare a research paper on the Linux operating system. In what ways does it support Stallman's "free as in freedom" doctrine for software? In what ways does Linux conflict with this philosophy?

1.13 Prepare a research paper on the Internet and how its pervasive use affects operating system design.

1.14 Prepare a research paper on the open-source software movement. Discuss whether all open-source software is free, as in both "freedom" and "price." How do the GPL and similar licenses promote open-source software?

1.15 Prepare a research paper on the evolution of operating systems. Be sure to mention the key hardware, software and communications technologies that encouraged each new operating system innovation.

1.16 Prepare a research paper on the future of operating systems.

1.17 Prepare a research paper giving a thorough taxonomy of past and present operating systems.

1.18 Prepare a research paper on Web services. Discuss the key technologies on which the Web services infrastructure is being built. How will the availability of Web services affect applications development?

1.19 Prepare a research paper on business-critical and mission-critical applications. Discuss the key attributes of hardware, communications software and operating systems that are essential to building systems to support these types of applications.

1.20 Prepare a research paper on virtual machine systems. Be sure to investigate IBM's VM operating system and Sun's Java Virtual Machine (JVM).

1.21 Prepare a research paper on operating systems and the law. Survey legislation related to operating systems.

1.22 Prepare a research paper on the impact of operating systems on business and the economy.

1.23 Prepare a research paper on operating systems and security and privacy. Be sure to consider the issues of worms and viruses.

1.24 Prepare a research paper on the ethical issues with which operating systems designers must be concerned. Be sure to deal with issues such as the use of computer systems in warfare and in life-threatening situations, viruses and worms, and other important topics you discover as you do your research for your paper.

1.25 List several trends leading the way to future operating systems designs. How will each affect the nature of future systems?

1.26 Prepare a research paper discussing the design of massively parallel systems. Be sure to compare large-scale multiprocessor systems (e.g., the Hewlitt-Packard Superdome supercomputer, which contains up to 64 processors; `www.hp.com/products1/servers/scalableservers/superdome/`) to clustered systems and server farms that contain hundreds or thousands of low-end computers that cooperate to perform common tasks (see, for example, `www.beowulf.org`). Use `www.top500.org`, a listing of the world's most powerful supercomputers, to determine the type of tasks that each of these massively parallel systems performs.

1.27 What trends are leading the way to dramatic increases in parallel computation? What challenges must be addressed by hardware designers and software designers before parallel computation will become widely used?

1.28 Prepare a research paper that compares MIT's Exokernel (`www.pdos.lcs.mit.edu/exo.html`) and CMU's Mach microkernel (`www-2.cs.cmu.edu/afs/cs.cmu.edu/project/mach/public/www/mach.html`) research operating systems.[119] What is the primary focus of each operating system? Be sure to mention how the researchers organized components such as memory management, disk scheduling and process management. Has either or both of these systems become commercially successful? Has either or both of these systems influenced the designs of commercially successful operating systems?

1.29 Why have UNIX and UNIX-based systems continued to be popular in recent decades? How does Linux impact the future of UNIX systems?

Chapter 2

Hardware and Software Concepts

Chapter 2

Exercise Solutions

2.1 Distinguish among hardware, software and firmware.

Ans: Hardware consists of the devices of the computer system including its processors, its storage, its I/O devices and its communication connections. Software consists of sets of instructions that programmers write to be executed by the hardware. There are two major software design principles—procedural programming and object-oriented programming. Firmware consists of persistently stored microcode instructions that perform the tasks specified by machine language. Firmware specifies software instructions but is part of the hardware.

2.2 Some hardware devices follow.

i. mainboard
ii. processor
iii. bus
iv. memory
v. hard disk
vi. peripheral device
vii. tertiary storage device
viii. register
ix. cache

Indicate which of these devices is best defined by each of the following. (Some items can have more than one answer.)

a. executes program instructions

Ans: Processor.

b. not required for a computer to execute program instructions.

Ans: Peripheral device (also tertiary storage device).

c. volatile storage medium.

Ans: Memory, cache and register.

d. the PCB that connects a system's processors to memory, secondary storage and peripheral devices.

Ans: Mainboard.

e. persistent storage medium

Ans: Hard disk and tertiary storage device (also some peripheral devices).

f. fastest memory in a computer system

Ans: Register.

g. set of traces that transmit data between hardware devices

Ans: Bus.

h. fast memory that improves application performance.

Ans: Cache.

i. lowest level of memory in the memory hierarchy that a processor can reference directly.

Ans: Memory.

2.3 Processor speeds have doubled roughly every 18 months. Has overall computer performance doubled at the same rate? Why or why not?

Ans: Although processor speeds have increased at an astonishing rate, slower development of memory technologies prevents overall computer performance from increasing at the same rate. Many systems are becoming I/O bound, meaning that processors must idle while they wait for I/O devices to complete transfers.

2.4 Sort the following list from fastest and most expensive memory to cheapest and least expensive memory: secondary storage, registers, main memory, tertiary storage, L2 cache, L1 cache.

Ans: Register memory, L1 cache, L2 cache, main memory, secondary storage, tertiary storage.

Why do systems contain several data stores of different size and speed?

Ans: The faster a type of memory is, the more expensive it is. It would nice to be able to fill all of system's memory needs with the fastest type of memory (registers), but this would be much too expensive (and impractical). It would also be nice to include a lot of memory in the system cheaply, but using only the slowest type of memory (tertiary storage) would make a system much too slow. Therefore, systems include many different types of memory so that the data accessed most-often is stored in the faster type of memory, increasing performance while still being able to store large quantities of data (using the slower types of memory). Also, faster memory is volatile, so a system must include some type of permanent storage so that all data is not lost.

What is the motivation behind caching?

Ans: Caching is a mechanism that speeds memory access by keeping duplicate copies of data (that the system thinks will be accessed in the future) in fast memory. Caches are used so that systems can store a large volume of data, but still enjoy low memory access times.

2.5 What are some costs and benefits of using nonvolatile RAM in all caches and main memory?

Ans: Nonvolatile RAM would most likely be more expensive than volatile RAM, and possibly a little slower. However, it would greatly benefit servers that need to be able to recover from power outages quickly, because the system could return to the state it was in before the outage.

2.6 Why is it important to support legacy architectures?

Ans: It allows software that has been written for previous architectures to be reused.

2.7 Relate the principle of least privilege to the concepts of user mode, kernel mode and privileged instructions.

Ans: The principle of least privilege states that programs should only have the level of privilege they need to carry out their functions, and no more. This prevents accidental or malicious corruption of the system. One of the simplest implementations of the principle of least privilege is the user mode/kernel mode dichotomy. Users execute in the user mode; the operating system executes in the kernel mode. Privileged instructions are executable only from the kernel mode, so the user, for example, cannot perform arbitrary I/O or access unprotected memory.

2.8 Describe several techniques for implementing memory protection.

Ans: Bounds registers may be used to specify a continuous range of addresses a program may access. Out-of-bounds references are not allowed. Virtual memory provides memory protection as well, because processes are unaware of the physical addresses to which the virtual addresses correspond. Therefore, a process may not access a physical address without the permission of the operating system.

2.9 Double buffering is a technique that allows an I/O channel and a processor to operate in parallel. On double-buffered input, for example, while a processor consumes one set of data in one buffer, the channel reads the next set of data into the other buffer so that the data will (hopefully) be ready for the processor. Explain in detail how a triple-buffering scheme might operate.

Ans: Consider a triple-buffered input scheme. First the channel reads data into buffer1. As the processor processes the data in buffer1, the channel proceeds to read data into buffer2 so that the data will (hopefully) be ready for the processor when the processor needs it. When the channel finishes reading data into buffer2, it begins reading data into buffer3, attempting to stay ahead of the processor. Eventually, the channel will want to deposit more data, so it might need to wait until the processor is finished processing the data in buffer1. Occasionally, the processor might need to wait for the channel to deposit more data for it to work with. Thus, the depositing of data and the reading of the data proceed in circular fashion using buffer1, then buffer2, then buffer3. This cycle repeats until all the data has been processed.

In what circumstances would triple buffering be effective?

Ans: Buffers are usually relatively large areas of storage, so before installing a triple-buffering scheme, we would like to know that it is worthwhile. Clearly, if the processor is much faster than the channel in the triple-buffered input scheme, then the channel may never be able to get far enough ahead of the processor to take advantage of the third buffer. Ideally, multiple-buffering schemes are used to pick up the occasional slack when one unit gets slightly ahead of the other. This can result in significant performance improvement.

2.10 Describe two different techniques for handling the communications between a processor and devices.

Ans: The two techniques are interrupts and polling. With polling, the processor repeatedly checks to see if it must service the channel. If not, then the processor tests the next channel. With polling, the processor is essentially always in control, but it may waste time discovering that devices do not need its attention. With interrupts, the channel that needs service generates a hardware signal called an interrupt that immediately gets the processor's attention. The advantage is that no time is wasted on devices that do not need attention.

2.11 Explain how DMA improves system performance. and cycle stealing.

Ans: Although both processors and channels need to have access to a memory module, only one may control the bus at a time. Early systems transferred data a character or word at a time to memory and interrupted the processor after each transfer. These interrupts reduced the time during which a processor could execute program instructions. With DMA, the processor is interrupted only after a block of data has been transferred. Once an I/O operation is initiated, characters are transferred to memory on a cycle-stealing basis. The channel temporarily usurps the processor's path to storage while a character is being transferred, then the processor continues operation. Favoring the channel over the processor helps keep the I/O devices operating efficiently; this is particularly critical in interactive systems, where the I/O consists of interactive user communication.

2.12 Why is it appropriate for channel controllers to steal cycles from processors when accessing memory?

Ans: Favoring the channel over the processor improves resource utilization by keeping the I/O devices busy. It is also important in interactive systems, which require fast response times from I/O.

2.13 Explain the notion of spooling and why it is useful.

Ans: In spooling, a high-speed device like a disk is interposed between a running program and a low-speed I/O device. This prevents the program execution time from being limited by the speed of the I/O device. One of the most common types of spooling is print spooling. Instead of a program directing its output to a relatively slow mechanical printing device, output is placed in a disk file (spool file). The program can run at the speed of the disk and therefore finish more quickly. When the printer becomes available, it reads the data from the spool file. Because the disk is much faster than the printer, the spooling program can drive the printer at top speed while the original user program proceeds with new activities in parallel with the spooling (this is one of the most common forms of multitasking, especially on personal computers).

How, do you suppose, does an input spooling system designed to read punched cards from a card reader operate?

Ans: First the deck of punched cards is spooled to a disk file. Then, as the program executes, it reads the card images from the disk much faster than it could from the card reader. The presumption is that the input spooling is completed before the program executes, so that the card images will be available as the program needs them. In the interim, while the cards are being spooled to disk, the program is performing other tasks. Although punched cards are not used today, the spooling principle illustrated here is still valid.

2.14 Consider the following types of programming languages:

i. machine language

ii. assembly language

iii. high-level language

iv. object-oriented programming language

v. structured programming language

Indicate which of these categories is best defined by each of the following. (Some items can have more than one answer.)

a. focuses on manipulating things (nouns)

Ans: Object-oriented programming language.

b. requires a translator programmer to convert the code into something a specific processor can understand

Ans: All except machine language.

c. written using 1's and 0's

Ans: Machine language.

d. defines a disciplined approach to software development and focuses on actions (verbs).

Ans: Structured programming language.