

<<操作系统>>

图书基本信息

书名：<<操作系统>>

13位ISBN编号：9787121112515

10位ISBN编号：7121112515

出版时间：2010-8

出版时间：电子工业出版社

作者：（美）斯托林斯 著，蒲晓蓉，周伟 改编

页数：604

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

前言

This book is about the concepts, structure, and mechanisms of operating systems. Its purpose is to present, as clearly and completely as possible, the nature and characteristics of modern-day operating systems. This task is challenging for several reasons. First, there is a tremendous range and variety of computer systems for which operating systems are designed. These include single-user workstations and personal computers, medium-sized shared systems, large mainframe and supercomputers, and specialized machines such as real-time systems. The variety is not just in the capacity and speed of machines, but in applications and system support requirements. Second, the rapid pace of change that has always characterized computer systems continues with no letup. A number of key areas in operating system design are of recent origin, and re-search into these and other new areas continues. In spite of this variety and pace of change, certain fundamental concepts apply consistently throughout. To be sure, the application of these concepts depends on the current state of technology and the particular application requirements. The intent of this book is to provide a thorough discussion of the fundamentals of operating system design and to relate these to contemporary design issues and to current directions in the development of operating systems.

<<操作系统>>

内容概要

《操作系统:精髓与设计原理(第6版)(英文版)》是讲解操作系统的经典教材,全书不仅系统地讲述了操作系统的基本概念、原理和方法,而且以当代最流行的操作系统——Windows Vista、UNIX和Linux为例,全面清楚地展现了当代操作系统的本质和特点。

与教材配套的专用站点,为教师和学生理解书中内容,提供了及时、生动的材料。

《操作系统:精髓与设计原理(第6版)(英文版)》既注重对操作系统经典知识的讲解,又紧密结合当代最新的操作系统发展趋势,可作为大学计算机专业双语教材和参考书,也可供从事计算机专业研究方向的技术人员参考。

<<操作系统>>

作者简介

作者：（美国）斯托林斯（William Stallings）改编：蒲晓蓉 周伟

书籍目录

Chapter 1 Computer System Overview 1.1 Basic Elements 1.2 Processor Registers 1.3 Instruction Execution
1.4 Interrupts 1.5 The Memory Hierarchy 1.6 Cache Memory 1.7 I/O Communication Techniques 1.8
Recommended Reading and Web Sites 1.9 KeyTerms, Review Questions, and Problems Appendix 1A
Performance Characteristics of Two-Level Memory Appendix 1B Procedure Control Chapter 2 Operating
System Overview 2.1 Operating System Objectives and Functions 2.2 The Evolution of Operating Systems 2.3
Major Achievements 2.4 Developments Leading to Modern Operating Systems 2.5 Microsoft Windows
Overview 2.6 Traditional UNIX Systems 2.7 Modern UNIX Systems 2.8 Linux 2.9 Recommended Reading
and Web Sites 2.10 Key Terms, Review Questions, and Problems Chapter 3 Process Description and Control 3.1
What is a Process ?
3.2 Process States 3.3 Process Description 3.4 Process Control 3.5 Execution of the Operating System 3.6
Security Issues 3.7 UNIX SVR4 Process Management 3.8 Summary 3.9 Recommended Reading 3.10
KeyTerms, Review Questions, and Problems Programming Project One Developing a Shell Chapter 4 Threads,
SMP, and Microkernels 4.1 Processes andThreads 4.2 Symmetric Multiprocessing (SMP) 4.3 Microkernels
4.4 WindowsVista Thread and SMP Management 4.5 Solaris Thread and SMP Management 4.6 Linux Process
andThread Management 4.7 Summary 4.8 Recommended Reading 4.9 KeyTerms, Review Questions, and
Problems Chapter 5 Concurrency: Mutual Exclusion and Synchronization 5.1 Principles of Concurrency 5.2
Mutual Exclusion: Hardware Support 5.3 Semaphores 5.4 Monitors 5.5 Message Passing 5.6 Readers/Writers
Problem 5.7 Summary 5.8 Recommended Reading 5.9 Key Terms, Review Questions, and Problems Chapter 6
Concurrency: Deadlock and Starvation 6.1 Principles of Deadlock 6.2 Deadlock Prevention 6.3 Deadlock
Avoidance 6.4 Deadlock Detection 6.5 An Integrated Deadlock Strategy 6.6 Dining Philosophers Problem 6.7
UNIX Concurrency Mechanisms 6.8 Linux Kernel Concurrency Mechanisms 6.9 Solaris Thread
Synchronization Primitives 6.10 Windows Vista Concurrency Mechanisms 6.11 Summary 6.12 Recommended
Reading 6.13 Key Terms, Review Questions, and Problems Chapter 7 Memory Management 7.1 Memory
Management Requirements 7.2 Memory Partitioning 7.3 Paging 7.4 Segmentation 7.5 Security Issues 7.6
Summary 7.7 Recommended Reading 7.8 Key Terms, Review Questions, and Problems Appendix 7A Loading
and Linking Chapter 8 Virtual Memory 328Chapter 9 Uniprocessor Scheduling 386Chapter 10 Multiprocessor
and Real-Time Scheduling Chapter 11 I/O Management and Disk Scheduling Chapter 12 File Management
Glossary References

章节摘录

插图：From the processor's point of view, the action for input is as follows. The processor issues a READ command. It then saves the context (e. g., program counter and processor registers) of the current program and goes off and does something else (e. g., the processor may be working on several different programs at the same time) . At the end of each instruction cycle, the processor checks for interrupts (Figure 1.7) . When the interrupt from the I/O module occurs, the processor saves the context of the program it is currently executing and begins to execute an interrupt-handling program that processes the interrupt. In this case, the processor reads the word of data from the I/O module and stores it in memory. It then restores the context of the program that had issued the I/O command (or some other program) and resumes execution. Figure 1.19b shows the use of interrupt-driven I/O for reading in a block of data. Interrupt-driven I/O is more efficient than programmed I/O because it eliminates needless waiting. However, interrupt-driven I/O still consumes a lot of processor time, because every word of data that goes from memory to I/O module or from I/O module to memory must pass through the processor. Almost invariably, there will be multiple I/O modules in a computer system, some mechanisms are needed to enable the processor to determine which device caused the interrupt and to decide, in the case of multiple interrupts, which one to handle first. In some systems, there are multiple interrupt lines, so that each I/O module signals on a different line. Each line will have a different priority. Alternatively, there can be a single interrupt line, but additional lines are used to hold a device address. Again, different devices are assigned different priorities.

<<操作系统>>

编辑推荐

《操作系统:精髓与设计原理(第6版)(英文版)》是畅销书作者William Stallings的力作,其第四版曾获得美国计算机科学与工程类教材大奖。

《操作系统:精髓与设计原理(第6版)(英文版)》清晰、完整地讲解了现代操作系统的概念、结构和机制。

这个新版本进行了完整的更新,以反映行业的最新进展。

作者全面分析了操作系统设计的高级目标和挑战,然后在每个主要领域给出了权威的讲解:线程、并发性、内存管理、虚拟内存、处理器调度、输入/输出管理以及文件管理等。

对每个领域,都用现实世界的例子进行了演示,从而强化了这些讨论。

这些例子基于Windows Vista、Linux、Unix以及其他同时期的操作系统。

不管是研究人员还是专业读者,《操作系统:精髓与设计原理(第6版)(英文版)》都是理想的读物。

全书讲解清晰、结构合理,并包含大量的教学支持材料,包括数百个经过仔细构思的实践性问题。

主要特点为了使复杂的概念更清晰、更易理解,集成了新的动画项目支持,包括两个手工编程项目、研究项目、阅读/撰写报告任务以及写作任务。配套网站WilliarnStallings.COIT/OS/OS6e.html上的大量Web内容支持丰富的关键字/缩略词清单、推荐读物、词汇表等涵盖内容大量的Windows例子,针对Windows Vista系统进行了全面更新扩充并优化了并发性内容全面讲解进程控制、线程、SMP和微内核对单处理器、多处理器和实时调度进行了详细探讨全面介绍如何管理内存、I/O、磁盘和文件

<<操作系统>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>