PREFACE

This book does not pretend to be a comprehensive record; but it aims at helping to disentangle from an immense mass of material the crucial issues and cardinal decisions. Throughout I have set myself to explain faithfully and to the best of my ability.

—*THE WORLD CRISIS*, WINSTON CHURCHILL

OBJECTIVES

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 embedded systems, smart phones, 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 as well. 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 research 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.

EXAMPLE SYSTEMS

This text is intended to acquaint the reader with the design principles and implementation issues of contemporary operating systems. Accordingly, a purely conceptual or theoretical treatment would be inadequate. To illustrate the concepts and to tie them to real-world design choices that must be made, three operating systems have been chosen as running examples:

- **Windows 7:** A multitasking operating system for personal computers, workstations, and servers. This operating system incorporates many of the latest developments in operating system technology. In addition, Windows is one of the first important commercial operating systems to rely heavily on

object-oriented design principles. This book covers the technology used in the most recent version of Windows, known as Windows 7.

- **UNIX:** A multiuser operating system, originally intended for minicomputers, but implemented on a wide range of machines from powerful microcomputers to supercomputers. Several flavors of UNIX are included as examples. FreeBSD is a widely used system that incorporates many state-of-the-art features. Solaris is a widely used commercial version of UNIX.
- **Linux:** An open-source version of UNIX that is now widely used.

These systems were chosen because of their relevance and representativeness. The discussion of the example systems is distributed throughout the text rather than assembled as a single chapter or appendix. Thus, during the discussion of concurrency, the concurrency mechanisms of each example system are described, and the motivation for the individual design choices is discussed. With this approach, the design concepts discussed in a given chapter are immediately reinforced with real-world examples.

INTENDED AUDIENCE

The book is intended for both an academic and a professional audience. As a textbook, it is intended as a one-semester undergraduate course in operating systems for computer science, computer engineering, and electrical engineering majors. It covers all of the core topics and most of the elective topics recommended in *Computer Science Curriculum 2008*, from the Joint Task Force on Computing Curricula of the IEEE Computer Society and the ACM, for the Undergraduate Program in Computer Science. The book also covers the operating systems topics recommended in the *Guidelines for Associate-Degree Curricula in Computer Science 2002*, also from the Joint Task Force on Computing Curricula of the IEEE Computer Society and the ACM. The book also serves as a basic reference volume and is suitable for self-study.

PLAN OF THE TEXT

The book is divided into eight parts (see Chapter 0 for an overview):

- Background
- Processes
- Memory
- Scheduling
- Input/output and files
- Embedded systems
- Security
- Distributed systems

The book includes a number of pedagogic features, including the use of animations and numerous figures and tables to clarify the discussion. Each chapter includes a list of key words, review questions, homework problems, suggestions for further reading, and recommended Web sites. The book also includes an extensive glossary, a list of frequently used acronyms, and a bibliography. In addition, a test bank is available to instructors.

WHAT'S NEW IN THE SEVENTH EDITION

In the 3 years since the sixth edition of this book was published, the field has seen continued innovations and improvements. In this new edition, I try to capture these changes while maintaining a broad and comprehensive coverage of the entire field. To begin the process of revision, the sixth edition of this book was extensively reviewed by a number of professors who teach the subject and by professionals working in the field. The result is that, in many places, the narrative has been clarified and tightened, and illustrations have been improved. Also, a number of new "field-tested" homework problems have been added.

Beyond these refinements to improve pedagogy and user friendliness, the technical content of the book has been updated throughout, to reflect the ongoing changes in this exciting field, and the instructor and student support has been expanded. The most noteworthy changes are as follows:

- **Windows 7:** Windows 7 is Microsoft's latest OS offering for PCs, workstations, and servers. The seventh edition provides details on Windows 7 internals in all of the key technology areas covered in this book, including process/thread management, scheduling, memory management, security, file systems, and I/O.
- **Multicore operating system issues:** The seventh edition now includes coverage of what has become the most prevalent new development in computer systems: the use of multiple processors on a single chip. At appropriate points in the book, operating system issues related to the use of a multicore organization are explored.
- **Virtual machines:** Chapter 2 now includes a section on virtual machines, which outlines the various approaches that have been implemented commercially.
- **New scheduling examples:** Chapter 10 now includes a discussion of the FreeBSD scheduling algorithm, designed for use with multiprocessor and multicore systems, and Linux VServer scheduling for a virtual machine environment.
- **Service-oriented architecture (SOA):** SOA is a form of client/server architecture that now enjoys widespread use in enterprise systems. SOA is now covered in Chapter 16.
- **Probability, statistics, and queueing analysis:** Two new chapters review key topics in these areas to provide background for OS performance analysis.
- **B-trees:** This is a technique for organizing indexes into files and databases that is commonly used in OS file systems, including those supported by

Mac OS X, Windows, and several Linux file systems. B-trees are now covered in Chapter 12.

- **Student study aids:** Each chapter now begins with a list of learning objectives. In addition, a chapter-by-chapter set of **review outlines** highlights key concepts that the student should concentrate on in each chapter.
- **OS/161:** OS/161 is an educational operating system that is becoming increasingly recognized as the teaching platform of choice. This new edition provides support for using OS/161 as an active learning component. See later in this Preface for details.
- **Sample syllabus:** The text contains more material than can be conveniently covered in one semester. Accordingly, instructors are provided with several sample syllabi that guide the use of the text within limited time (e.g., 16 weeks or 12 weeks). These samples are based on real-world experience by professors with the sixth edition.

With each new edition, it is a struggle to maintain a reasonable page count while adding new material. In part, this objective is realized by eliminating obsolete material and tightening the narrative. For this edition, chapters and appendices that are of less general interest have been moved online, as individual PDF files. This has allowed an expansion of material without the corresponding increase in size and price.

STUDENT RESOURCES

For this new edition, a tremendous amount of original supporting material has been made available online, in the following categories
The Companion Web site and student resource material can be reached through the Publisher's Web site www.pearsonhighered.com/stallings or by clicking on the button labeled "Book Info and More Instructor Resources" at the book's Companion Web site WilliamStallings.com/OS/OS7e.html. For this new edition, a tremendous amount of original supporting material has been made available online, in the following categories:

- **Homework problems and solutions:** To aid the student in understanding the material, a separate set of homework problems with solutions are available. These enable the students to test their understanding of the text.
- **Programming projects:** Two major programming projects, one to build a shell (or command line interpreter) and one to build a process dispatcher, are described.
- **Key papers:** Several dozen papers from the professional literature, many hard to find, are provided for further reading.
- **Supporting documents:** A variety of other useful documents are referenced in the text and provided online.

Premium Web Content
Purchasing this textbook new grants the reader 6 months of access to this online material. See the access card in the front of this book for details.

- **Online chapters:** To limit the size and cost of the book, four chapters of the book are provided in PDF format. The chapters are listed in this book's table of contents.
- **Online appendices:** There are numerous interesting topics that support material found in the text but whose inclusion is not warranted in the printed text. A total of 13 appendices cover these topics for the interested student. The appendices are listed in this book's table of contents.

INSTRUCTOR SUPPORT MATERIALS

Support materials are available at the Instructor Resource Center (IRC) for this textbook, which can be reached through the Publisher's Web site www.pearsonhighered.com/stallings or by clicking on the button labeled "Book Info and More Instructor Resources" at this book's Companion Web site WilliamStallings.com/OS/OS7e.html. To gain access to the IRC, please contact your local Pearson sales representative via pearsonhighered.com/educator/replocator/requestSalesRep.page or call Pearson Faculty Services at 1-800-526-0485. To support instructors, the following materials are provided:

- **Solutions manual:** Solutions to end-of-chapter Review Questions and Problems.
- **Projects manual:** Suggested project assignments for all of the project categories listed in the next section.
- **PowerPoint slides:** A set of slides covering all chapters, suitable for use in lecturing.
- **PDF files:** Reproductions of all figures and tables from the book.
- **Test bank:** A chapter-by-chapter set of questions.
- Links to Web sites for other courses being taught using this book.
- An Internet mailing list has been set up so that instructors using this book can exchange information, suggestions, and questions with each other and with the author. As soon as typos or other errors are discovered, an errata list for this book will be available at WilliamStallings.com. Sign-up information for this Internet mailing list.
- **Computer science student resource list:** A list of helpful links for computer science students and professionals is provided at ComputerScienceStudent.com, which provides documents, information, and useful links for computer science students and professionals.
- **Programming projects:** Two major programming projects, one to build a shell (or command line interpreter) and one to build a process dispatcher, are described in the online portion of this textbook. The IRC provides further information and step-by-step exercises for developing the programs. As an alternative, the instructor can assign a more extensive series of projects that cover many of the principles in the book. The student is provided with

detailed instructions for doing each of the projects. In addition, there is a set of homework problems, which involve questions related to each project for the student to answer.

Projects and Other Student Exercises

For many instructors, an important component of an OS course is a project or set of projects by which the student gets hands-on experience to reinforce concepts from the text. This book provides an unparalleled degree of support for including a projects component in the course. In the online portion of the text, two major programming projects are defined. In addition, the instructor support materials available through Pearson not only include guidance on how to assign and structure the various projects but also includes a set of user's manuals for various project types plus specific assignments, all written especially for this book. Instructors can assign work in the following areas:

- **OS/161 projects:** Described below.
- **Simulation projects:** Described below.
- **Programming projects:** Described below.
- **Research projects:** A series of research assignments that instruct the student to research a particular topic on the Internet and write a report.
- **Reading/report assignments:** A list of papers that can be assigned for reading and writing a report, plus suggested assignment wording.
- **Writing assignments:** A list of writing assignments to facilitate learning the material.
- **Discussion topics:** These topics can be used in a classroom, chat room, or message board environment to explore certain areas in greater depth and to foster student collaboration.

In addition, information is provided on a software package known as BACI that serves as a framework for studying concurrency mechanisms.

This diverse set of projects and other student exercises enables the instructor to use the book as one component in a rich and varied learning experience and to tailor a course plan to meet the specific needs of the instructor and students. See Appendix B in this book for details.

OS/161

New to this edition is support for an active learning component based on OS/161. OS/161 is an educational operating system that is becoming increasingly recognized as the preferred teaching platform for OS internals. It aims to strike a balance between giving students experience in working on a real operating system and potentially overwhelming students with the complexity that exists in a fully fledged operating system, such as Linux. Compared to most deployed operating systems, OS/161 is quite small (approximately 20,000 lines of code and comments), and therefore it is much easier to develop an understanding of the entire code base.

The IRC includes:

1. A packaged set of html files that the instructor can upload to a course server for student access.
2. A getting-started manual to be handed out to students to help them begin using OS/161.
3. A set of exercises using OS/161, to be handed out to students.
4. Model solutions to each exercise for the instructor's use.
5. All of this will be cross-referenced with appropriate sections in the book, so that the student can read the textbook material and then do the corresponding OS/161 project.

Simulations for Students and Instructors

The IRC provides support for assigning projects based on a set of seven **simulations** that cover key areas of OS design. The student can use a set of simulation packages to analyze OS design features. The simulators are all written in Java and can be run either locally as a Java application or online through a browser. The IRC includes specific assignments to give to students, telling them specifically what they are to do and what results are expected.

Animations for Students and Instructors

This edition also incorporates animations. Animations provide a powerful tool for understanding the complex mechanisms of a modern OS. A total of 53 animations are used to illustrate key functions and algorithms in OS design. The animations are used for Chapters 3, 5, 6, 7, 8, 9, and 11. For access to the animations, click on the rotating globe at this book's Web site at WilliamStallings.com/OS/OS7e.html.

ACKNOWLEDGMENTS

This new edition has benefited from review by a number of people, who gave generously of their time and expertise. These include Samir Chettri (The University of Maryland, Baltimore County), Michael Rogers (Tennessee Technological University), Glenn Booker (Drexel University), Jeongkyu Lee (University of Bridgeport), Sanjiv Bhatia (University of Missouri, Baltimore County), Martin Barrett (East Tennessee State University), Lubomir Ivanov (Iona College), Bina Ramamurthy (University at Buffalo), Dean Kelley (Minnesota State University), Joel Weinstein (Northeastern University), all of whom reviewed most or all of the book.

Thanks also to the people who provided detailed reviews of a one or more chapters: John South (University of Dallas), Kevin Sanchez-Cherry (IT Security Specilist), Adri Jovin (PG Scholar,Department of IT, Anna University of Technology, Coimbatore), Thriveni Venkatesh (Professor Thriveni T K from GcMAT, Bangalore, India), Fernando Lichtschein (Instituto de Tecnología ORT Argentina), C. Annamala (Indian Institute of Technology Kharagpur),

Abdul-Rahman Mahmood (Independent IT & security consultant & Creator of AlphaPeeler crypto tool), and Abhilash V R (VVDN Technologies).

I would also like to thank Dave Probert, Architect in the Windows Core Kernel & Architecture team at Microsoft, for the review of the material on Windows Vista and for providing the comparisons of Linux and Vista; Tigran Aivazian, author of the Linux Kernel Internals document, which is part of the Linux Documentation Project, for the review of the material on Linux 2.6; Nick Garnett of eCosCentric, for the review of the material on eCos; and Philip Levis, one of the developers of TinyOS, for the review of the material on TinyOS.

Professor Andrew Peterson, Ioan Stefanovici, and OS instructors at the University of Toronto prepared the OS/161 supplements for the IRC.

Adam Critchley (University of Texas at San Antonio) developed the simulation exercises. Matt Sparks (University of Illinois at Urbana-Champaign) adapted a set of programming problems for use with this textbook.

Lawrie Brown of the Australian Defence Force Academy produced the material on buffer overflow attacks. Ching-Kuang Shene (Michigan Tech University) provided the examples used in the section on race conditions and reviewed the section. Tracy Camp and Keith Hellman, both at the Colorado School of Mines, developed a new set of homework problems. In addition, Fernando Ariel Gont contributed a number of homework problems; he also provided detailed reviews of all of the chapters.

I would also like to thank Bill Bynum (College of William and Mary) and Tracy Camp (Colorado School of Mines) for contributing Appendix O; Steve Taylor (Worcester Polytechnic Institute) for contributing the programming projects and reading/report assignments in the instructor's manual; and Professor Tan N. Nguyen (George Mason University) for contributing the research projects in the instruction manual. Ian G. Graham (Griffith University) contributed the two programming projects in the textbook. Oskars Rieksts (Kutztown University) generously allowed me to make use of his lecture notes, quizzes, and projects.

Finally, I would like to thank the many people responsible for the publication of the book, all of whom did their usual excellent job. This includes my editor Tracy Dunkelberger, her assistants Carole Snyder, Melinda Hagerty, and Allison Michael, and production manager Pat Brown. I also thank Shiny Rajesh and the production staff at Integra for another excellent and rapid job. Thanks also to the marketing and sales staffs at Pearson, without whose efforts this book would not be in your hands.

With all this assistance, little remains for which I can take full credit. However, I am proud to say that, with no help whatsoever, I selected all of the quotations.