

# Lecture Note 0: Course Introduction

February 25, 2021  
Jongmoo Choi

Dept. of software  
Dankook University

<http://embedded.dankook.ac.kr/~choijm>

# Who am I?

## Lecture site

The screenshot shows the home page of Jongmoo Choi. At the top, it says "Jongmoo Choi's Home Page" and "embedded.dankook.ac.kr/~choijm/". Below the header, there is a photo of Professor Choi and his contact information: "Professor (Ph.D.)", "Dept. of Software (Dept. of Computer Science and Engineering) Dankook University", "152, Jukjeon-ro, Suji-gu, Yongin-si, Gyeonggi-do, Korea, 16890 (126, Jukjeon-Dong, Suji-Gu, Yongin, Korea)", "Email: choijm (at) dankook (dot) ac (dot) kr", "Office: SW-ICT Building, Room 504", "Telephone: +82-31-8005-3242, 3247(lab)", "FAX: +82-31-8021-7220", and "연구실 홈페이지, Google Scholar". A "Contents" section lists: Curriculum Vitae, Publication Lists, Course Information, Projects, Photographs, Presentations, Personal Interests, and Weekly Schedule. A "Curriculum Vitae" section mentions his position at Dankook University from 2003 to the present and his time at Carnegie Mellon University in 2011 and 2015.

(home page)

The screenshot shows the "Course Information" page. It lists "Contents" for the "운영체제 (Operating System)" course: "2021년 1학기: 운영체제 (Operating System)", "2021년 1학기: IT 응용 II (IT Application II)", and "Previous Lecture Information". Under "운영체제 (Operating System)", it lists "강의 자료 (Lecture Notes)" with links for LN 0 through LN 9, and "강의 교재" (Main textbook: *Operating systems: Three Easy Pieces* by R. Arpaci-Dusseau and A. Arpaci-Dusseau). It also lists "강의 관련 자료" (References) including *Operating system concept (10th edition)*, *Operating Systems: Internals and Design Principles (9th edition)*, *Modern Operating Systems(4th edition)*, and *Linux Kernel Development(3th edition)*.

(lecture page)

The screenshot shows the e-learning campus interface. It displays "최근 공지사항" (Recent Notices) for the "운영체제(SW) 1분반의 최근 활동" (Recent activities of the OS(SW) 1st semester). A notice box says "최근 메시지 없음" (No recent messages). Below, there are sections for "과목 상태" (Course Status) with buttons for "비공개" (Private) and "공개" (Public), and "사용자 및 그룹" (Users and Groups) with a "다시 공지" (Re-notify) button. Other sections include "성적" (Grade) with "학생 화면 보기" (View student screen), "종합성적부" (Comprehensive grade book) with "과목 분석 보기" (View subject analysis), "파일" (File) with "과목 알림 보기" (View subject notification), "페이지" (Page), "성과" (Performance) with "행정" (Administration) and "일람 보기" (View list), "진도" (Progress) with "다음주까지 없음" (None until next week), and "합업" (Collaboration).

(e-learning campus)

# What is Operating System?

## ■ Definition (from wikipedia.org)

From Wikipedia, the free encyclopedia

An **operating system (OS)** is system software that manages computer hardware and software resources and provides common services for computer programs.

Time-sharing operating systems schedule tasks for efficient use of the system and may also include accounting software for cost allocation of processor time, mass storage, printing, and other resources.

For hardware functions such as input and output and memory allocation, the operating system acts as an intermediary between programs and the computer hardware,<sup>[1][2]</sup> although the application code is usually executed directly by the hardware and frequently makes system calls to an OS function or is interrupted by it. Operating systems are found on many devices that contain a computer – from cellular phones and video game consoles to web servers and supercomputers.

The dominant desktop operating system is Microsoft Windows with a market share of around 82.74%. macOS by Apple Inc. is in second place (13.23%), and the varieties of Linux are collectively in third place (1.57%).<sup>[3]</sup> In the mobile (smartphone and tablet combined) sector, use in 2017 is up to 70% of Google's Android<sup>[4]</sup> and according to third quarter 2016 data, Android on smartphones is dominant with 87.5 percent and a growth rate 10.3 percent per year, followed by Apple's iOS with 12.1 percent and a per year decrease in market share of 5.2 percent, while other operating systems amount to just 0.3 percent.<sup>[5]</sup> Linux distributions are dominant in the server and supercomputing sectors. Other specialized classes of operating systems, such as embedded and real-time systems, exist for many applications.

**Operating systems**

User

Application

Operating System

Hardware

**Common features**

- Process management · Interrupts ·
- Memory management · File system ·
- Device drivers · Networking · Security · I/O

V · T · E

# Course Objectives

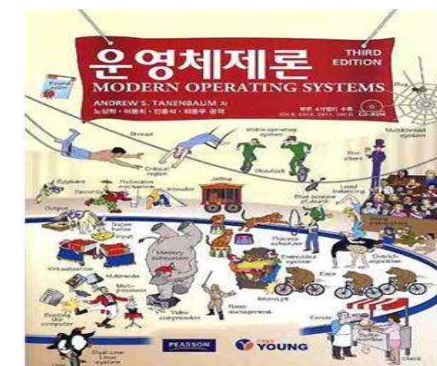
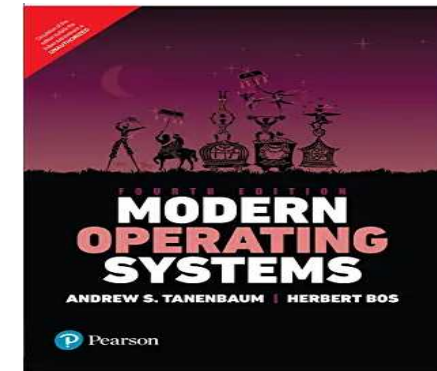
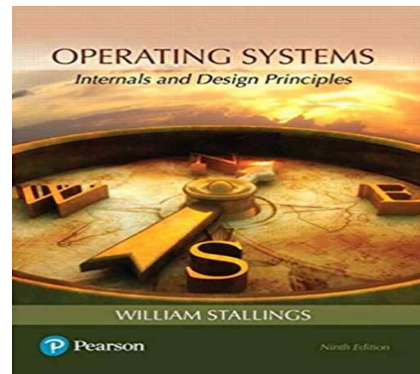
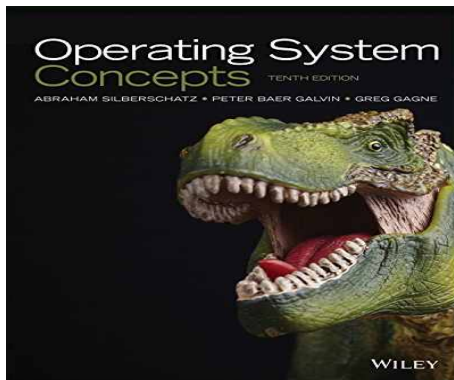
---

- Understand the **definition**, role and goal of OS
  - ✓ Resource manager, computing environments, ...
- Know the existing operating systems
  - ✓ UNIX, Windows, Apple OS X, Linux, Android, iOS, WebOS, Mach, ...
- Learn the **internal structure** of OS
  - ✓ Process, Virtual memory, File system, Driver, Protocol, Interrupt, ...
- Comprehend the **policies** and **mechanisms** used by OS
  - ✓ CPU scheduling, Demand paging, LRU, inode, System call, ...
- Grasp the idea of abstraction
  - ✓ Information Hiding, Illusion, Interface, Layered architecture, ...
- **Demonstrate** what we have learned
  - ✓ Lab. project



# Traditional Textbook

- Three representative textbooks for operating system course
  - ✓ Operating Systems Concepts (10<sup>th</sup> edition), by A. Silberschatz, P. Galvin and G. Gagne
  - ✓ Operating Systems: Internals and Design Principles (9<sup>th</sup> edition), by W. Stalling
  - ✓ Modern Operating Systems (5<sup>th</sup> edition), by A. Tanenbaum and H. Bos



# Textbook in this course

## ■ Remzi's OSTEP (OS Three Easy Pieces)

✓ <http://pages.cs.wisc.edu/~remzi/OSTEP/>

Operating Systems: Three Easy | x +  
 pages.cs.wisc.edu/~remzi/OSTEP/  
 Another way to help the book out: cite it! Here is the [BiBTeX entry \(seen below\)](#); you can also link to the site on the market.  
**Operating Systems: Three Easy Pieces**  
 Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau  
 Arpaci-Dusseau Books  
 August, 2018 (Version 1.00)

Course Information  
 Contents  
 • 2021년 1학기: 운영체제 (Operating System)  
 • 2021년 1학기: IT 응용 II (IT Application II)  
 • Previous Lecture Information  
 운영체제 (Operating System)  
 • 강의 자료 (Lecture Notes)  
 o LN 0: PPT for Course overview  
 o LN 1: OS Introduction (OSTEP Chap.1-2)  
 o LN 2: Process (OSTEP Chap.3-6)  
 o LN 3: Scheduling (OSTEP Chap.7-11)  
 o LN 4: Thread and Lock (OSTEP Chap.25-29)  
 o LN 5: Semaphore and Deadlock (OSTEP Chap.30-32)  
 o LN 6: File system basic (OSTEP Chap.35-40)  
 o LN 7: Advanced File system (OSTEP Chap.41-42)  
 o LN 8: Memory Management (OSTEP Chap.12-17)  
 o LN 9: Paging and Beyond Physical Memory (OSTEP Chap.18-22)  
 • 강의 교재  
 o Main textbook: *Operating systems: Three Easy Pieces*, by R. Arpaci-Dusseau and A. Arpaci-Dusseau  
 • 강의 관련 자료  
 o Reference 1: *Operating system concept (10th edition)*, by A. Silberschatz, P. B. Galvin and G. Gagne, *Korean version*  
 o Reference 2: *Operating Systems: Internals and Design Principles (9th edition)*, by W. Stalling, *Korean version*  
 o Reference 3: *Modern Operating Systems(4th edition)*, by A. Tanenbaum, *Korean version*  
 o Reference 4: *Linux Kernel Development(3th edition)*, by R. Love, *Korean version*

And now, the free online form of the book, in chapter-by-chapter form (now with chapter numbers!):

| Intro                                | Virtualization                             | Concurrency                                      | Persistence                                | Security                 |
|--------------------------------------|--|--|--|--------------------------|
| Preface                              | 3 <i>Dialogue</i>                          | 25 <i>Dialogue</i>                               | 35 <i>Dialogue</i>                         | 52 <i>Dialogue</i>       |
| TOC                                  | 4 Processes                                | 26 <i>Concurrency and Threads</i><br><i>code</i> | 36 <i>I/O Devices</i>                      | 53 <i>Intro Security</i> |
| 1 <i>Dialogue</i>                    | 5 <i>Process API</i><br><i>code</i>        | 27 <i>Thread API</i><br><i>code</i>              | 37 <i>Hard Disk Drives</i>                 | 54 <i>Authentication</i> |
| 2 <i>Introduction</i><br><i>code</i> | 6 <i>Direct Execution</i>                  | 28 <i>Locks</i><br><i>code</i>                   | 38 <i>Redundant Disk Arrays (RAID)</i>     | 55 <i>Access Control</i> |
|                                      | 7 <i>CPU Scheduling</i>                    | 29 <i>Locked Data Structures</i>                 | 39 <i>Files and Directories</i>            | 56 <i>Cryptography</i>   |
|                                      | 8 <i>Multi-level Feedback</i>              | 30 <i>Condition Variables</i><br><i>code</i>     | 40 <i>File System Implementation</i>       | 57 <i>Distributed</i>    |
|                                      | 9 <i>Lottery Scheduling</i><br><i>code</i> | 31 <i>Semaphores</i><br><i>code</i>              | 41 <i>Fast File System (FFS)</i>           |                          |
|                                      | 10 <i>Multi-CPU Scheduling</i>             | 32 <i>Concurrency Bugs</i>                       | 42 <i>FSCK and Journaling</i>              | <b>Appendices</b>        |
|                                      | 11 <i>Summary</i>                          | 33 <i>Event-based Concurrency</i>                | 43 <i>Log-structured File System (LFS)</i> | <i>Dialogue</i>          |
|                                      |  | 34 <i>Summary</i>                                | 44 <i>Flash-based SSDs</i>                 | <i>Virtual Machines</i>  |
|                                      |  |  | 45 <i>Data Integrity and Protection</i>    | <i>Dialogue</i>          |
|                                      |  |  | 46 <i>Summary</i>                          | <i>Monitors</i>          |
|                                      |  |  | 47 <i>Dialogue</i>                         | <i>Dialogue</i>          |
|                                      |  |  | 48 <i>Distributed Systems</i>              | <i>Lab Tutorial</i>      |
|                                      |  |  | 49 <i>Network File System (NFS)</i>        | <i>Systems Labs</i>      |
|                                      |  |  | 50 <i>Andrew File System (AFS)</i>         | <i>xv6 Labs</i>          |
|                                      |  |  | 51 <i>Summary</i>                          |                          |

**INSTRUCTORS:** If you are using these free chapters, please just link to them directly (instead of making a copy locally); we make little improvements frequently and thus would like to provide the latest to whomever is using it. Also, we have made our own class preparation notes available to those of

# Textbook in this course

## ■ TOC (Table of Contents) of OSTEP

| Contents   |           |
|--|-----------|
| To Everyone . . . . .                                | iii       |
| To Educators . . . . .                               | v         |
| To Students . . . . .                                | vi        |
| Acknowledgments . . . . .                            | vii       |
| Final Words . . . . .                                | x         |
| References . . . . .                                 | xi        |
| <b>1 A Dialogue on the Book . . . . .</b>            | <b>1</b>  |
| <b>2 Introduction to Operating Systems . . . . .</b> | <b>3</b>  |
| 2.1 Virtualizing The CPU . . . . .                   | 5         |
| 2.2 Virtualizing Memory . . . . .                    | 7         |
| 2.3 Concurrency . . . . .                            | 8         |
| 2.4 Persistence . . . . .                            | 11        |
| 2.5 Design Goals . . . . .                           | 13        |
| 2.6 Some History . . . . .                           | 14        |
| 2.7 Summary . . . . .                                | 18        |
| References . . . . .                                 | 19        |
| Homework . . . . .                                   | 20        |
| <b>I Virtualization . . . . .</b>                    | <b>21</b> |
| <b>3 A Dialogue on Virtualization . . . . .</b>      | <b>23</b> |
| <b>4 The Abstraction: The Process . . . . .</b>      | <b>25</b> |
| 4.1 The Abstraction: A Process . . . . .             | 26        |
| 4.2 Process API . . . . .                            | 27        |
| 4.3 Process Creation: A Little More Detail . . . . . | 28        |
| 4.4 Process States . . . . .                         | 29        |
| 4.5 Data Structures . . . . .                        | 31        |
| 4.6 Summary . . . . .                                | 33        |
| References . . . . .                                 | 34        |

| CONTENTS  |           |
|---|-----------|
| Homework (Simulation) . . . . .                               | 35        |
| <b>5 Interlude: Process API . . . . .</b>                     | <b>37</b> |
| 5.1 The <code>fork()</code> System Call . . . . .             | 37        |
| 5.2 The <code>wait()</code> System Call . . . . .             | 39        |
| 5.3 Finally, The <code>exec()</code> System Call . . . . .    | 40        |
| 5.4 Why? Motivating The API . . . . .                         | 41        |
| 5.5 Process Control And Users . . . . .                       | 44        |
| 5.6 Useful Tools . . . . .                                    | 45        |
| 5.7 Summary . . . . .   | 45        |
| References . . . . .  | 47        |
| Homework (Code) . . . . .                                     | 48        |
| <b>6 Mechanism: Limited Direct Execution . . . . .</b>        | <b>49</b> |
| 6.1 Basic Technique: Limited Direct Execution . . . . .       | 49        |
| 6.2 Problem #1: Restricted Operations . . . . .               | 50        |
| 6.3 Problem #2: Switching Between Processes . . . . .         | 55        |
| 6.4 Worried About Concurrency? . . . . .                      | 59        |
| 6.5 Summary . . . . .   | 60        |
| References . . . . .  | 62        |
| Homework (Measurement) . . . . .                              | 63        |
| <b>7 Scheduling: Introduction . . . . .</b>                   | <b>65</b> |
| 7.1 Workload Assumptions . . . . .                            | 65        |
| 7.2 Scheduling Metrics . . . . .                              | 66        |
| 7.3 First In, First Out (FIFO) . . . . .                      | 66        |
| 7.4 Shortest Job First (SJF) . . . . .                        | 68        |
| 7.5 Shortest Time-to-Completion First (STCF) . . . . .        | 69        |
| 7.6 A New Metric: Response Time . . . . .                     | 70        |
| 7.7 Round Robin . . . . .                                     | 71        |
| 7.8 Incorporating I/O . . . . .                               | 73        |
| 7.9 No More Oracle . . . . .                                  | 74        |
| 7.10 Summary . . . . .  | 74        |
| References . . . . .  | 75        |
| Homework (Simulation) . . . . .                               | 76        |
| <b>8 Scheduling: The Multi-Level Feedback Queue . . . . .</b> | <b>77</b> |
| 8.1 MLFQ: Basic Rules . . . . .                               | 78        |
| 8.2 Attempt #1: How To Change Priority . . . . .              | 79        |
| 8.3 Attempt #2: The Priority Boost . . . . .                  | 83        |
| 8.4 Attempt #3: Better Accounting . . . . .                   | 84        |
| 8.5 Tuning MLFQ And Other Issues . . . . .                    | 84        |
| 8.6 MLFQ: Summary . . . . .                                   | 86        |
| References . . . . .  | 87        |
| Homework (Simulation) . . . . .                               | 88        |

# Reference

## ■ Linux Kernel Internals (리눅스 커널 내부 구조)

- ✓ 1장. 리눅스 소개
- ✓ 2장. 리눅스 커널 구조
- ✓ 3장. 태스크 관리
- ✓ 4장. 메모리 관리
- ✓ 5장. 파일시스템과 가상 파일시스템
- ✓ 6장. 인터럽트와 트랩 그리고 시스템 호출
- ✓ 7장. 리눅스 모듈 프로그래밍
- ✓ 8장. 디바이스 드라이버
- ✓ 9장. 네트워킹
- ✓ 10장. 운영체제 관련 실습
- ✓ 부록1. 리눅스와 가상화 그리고 XEN
- ✓ 부록2. MTD와 YAFFS





# Teaching Method

## ■ Mainly Lecturing

- ✓ Discussion (Q&A) during the course is quite important

## ■ Homework

- ✓ Reading assignment
  - 2 or 3 times
- ✓ Lab. Project (Programming or Analysis)
  - Lab1: scheduling
  - Lab2: concurrency
  - Lab3: file system



## ■ Grading

- ✓ Exam(45%) + Lab. Project (35%) + Assignment/Discussion (10%) + Attendance/Quiz/Discussion (10%)
- ✓ **Absence more than 5 times** or **Mid or Final Exam. score below 20** or **No lab. Project** → F
- ✓ Roughly, 20% students are expected to get the A grade.

# Discussion

---



➤ Any questions? Ask at “문의 게시판” or Send an email to me: [choijm@dankook.ac.kr](mailto:choijm@dankook.ac.kr)



# Quiz for 1<sup>th</sup>-Week 1<sup>st</sup>-Lesson

## ■ Quiz

- ✓ 1. What are the difference between Operating Systems (e.g. MS Windows or Linux) and Applications (e.g. MS Word or Chrome)? Explain the difference using the word “mode”.
- ✓ 2. Find out the philosopher who appears in Chapter 1, “A Dialog on the Book”, of the OSTEP (our main text book).
- ✓ Due: until 6 PM Friday of this week (5<sup>th</sup>, March)



(Source: Google Image)

Operating Systems: Three Easy

pages.cs.wisc.edu/~remzi/OSTEP/

Another way to help the book out: cite it! Here is the [BIBTeX entry \(seen below\)](#); you can also link to the site of the [best free operating systems book](#) on the market.

**Operating Systems: Three Easy Pieces**  
 Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau  
 Arpaci-Dusseau Books  
 August, 2018 (Version 1.00)

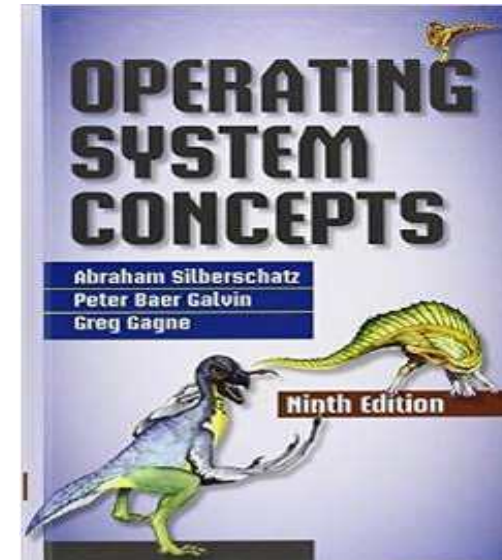
And now, the free online form of the book, in chapter-by-chapter form (now with chapter numbers!):

| Intro                      | Virtualization                         | Concurrency                         | Persistence                                  | Security                            |
|----------------------------|--|-------------------------------------|--|-------------------------------------|
| Preface                    | 3 <i>Dialogue</i>                      | 12 <i>Dialogue</i>                  | 35 <i>Dialogue</i>                           | 52 <i>Dialogue</i>                  |
| TOC                        | 4 Processes                            | 13 Address Spaces <code>code</code> | 26 Concurrency and Threads <code>code</code> | 43 <i>Intro_Security</i>            |
| 1 <i>Dialogue</i>          | 5 Process API <code>code</code>        | 14 Memory API                       | 27 Thread API <code>code</code>              | 37 Hard Disk Drives                 |
| 2 <i>Introduction code</i> | 6 Direct Execution                     | 15 Address Translation              | 28 Locks <code>code</code>                   | 38 Redundant Disk Arrays (RAID)     |
|                            | 7 CPU Scheduling                       | 16 Segmentation                     | 29 Locked Data Structures                    | 39 Files and Directories            |
|                            | 8 Multi-level Feedback                 | 17 Free Space Management            | 30 Condition Variables <code>code</code>     | 40 File System Implementation       |
|                            | 9 Lottery Scheduling <code>code</code> | 18 Introduction to Paging           | 31 Semaphores <code>code</code>              | 41 Fast File System (FFS)           |
|                            | 10 Multi-CPU Scheduling                | 19 Translation Lookaside Buffers    | 32 Concurrency Bugs                          | 42 FSCK and Journaling              |
|                            | 11 <i>Summary</i>                      | 20 Advanced Page Tables             | 33 Event-based Concurrency                   | 43 Log-structured File System (LFS) |
|                            |  | 21 Swapping: Mechanisms             | 34 <i>Summary</i>                            | 44 Flash-based SSDs                 |
|                            |  | 22 Swapping: Policies               |  | 45 Data Integrity and Protection    |
|                            |  | 23 Complete VM Systems              |  | 46 <i>Summary</i>                   |
|                            |  | 24 <i>Summary</i>                   |  | 47 <i>Dialogue</i>                  |
|                            |  |                                     |  | 48 Distributed Systems              |
|                            |  |                                     |  | 49 Network File System (NFS)        |
|                            |  |                                     |  | 50 Andrew File System (AFS)         |
|                            |  |                                     |  | 51 <i>Summary</i>                   |

**INSTRUCTORS:** If you are using these free chapters, please just link to them directly (instead of making a copy locally); we make little improvements frequently, and they would like to provide the latest to whomsoever uses it. Also, we have made our own class presentation notes available to those of

# Reference

- Operating Systems Concepts
  - ✓ Chapter 1: Introduction
  - ✓ Chapter 2: Operating System Structure
  - ✓ Chapter 3: Processes
  - ✓ Chapter 4: Threads
  - ✓ Chapter 5: Process Synchronization
  - ✓ Chapter 6: CPU Scheduling
  - ✓ Chapter 7: Deadlocks
  - ✓ Chapter 8: Main Memory
  - ✓ Chapter 9: Virtual Memory
  - ✓ Chapter 10: Mass-Storage Structure
  - ✓ Chapter 11: File System Interface
  - ✓ Chapter 12: File System Implementation
  - ✓ Chapter 13: I/O Systems
  - ✓ Chapter 14: Protection
  - ✓ Chapter 15: Security
  - ✓ Chapter 16: Virtual machine
  - ✓ Chapter 17: Distributed Systems
  - ✓ Chapter 18: The Linux System
  - ✓ Chapter 19: Windows 7
  - ✓ Chapter 20: Influential OSES
  - ✓ Appendix: Mach, BSD

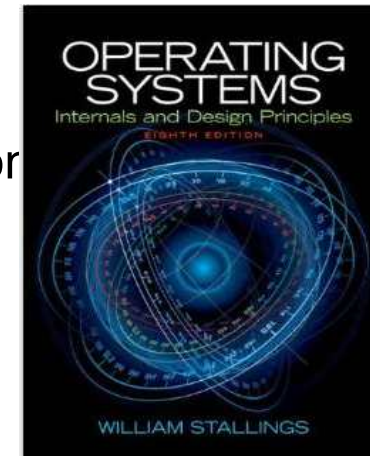


# Reference

---

## ■ Operating Systems: Internals and Design Principles

- ✓ Chapter 1. Computer system overview
- ✓ Chapter 2. Operating system overview
- ✓ Chapter 3. Process description and control
- ✓ Chapter 4. Threads
- ✓ Chapter 5. Concurrency: Mutual exclusion & synchronor
- ✓ Chapter 6. Concurrency: Deadlock & Starvation
- ✓ Chapter 7. Memory management
- ✓ Chapter 8. Virtual memory
- ✓ Chapter 9. Uniprocessor scheduling
- ✓ Chapter 10. Multiprocessor and RT scheduling
- ✓ Chapter 11. I/O management and disk scheduling
- ✓ Chapter 12. File management
- ✓ Chapter 13. Embedded operating system
- ✓ Chapter 14. Virtual machine
- ✓ Chapter 15. Operating system security
- ✓ Chapter 16. Distributed processing, Client/Server, and Cluster
- ✓ Appendix and Online chapter



# Reference

---

## ■ Modern Operating Systems

- ✓ Chapter 1. Introduction
- ✓ Chapter 2. Process and Thread
- ✓ Chapter 3. Memory Management
- ✓ Chapter 4. File Systems
- ✓ Chapter 5. Input/Output
- ✓ Chapter 6. Deadlocks
- ✓ Chapter 7. Virtualization and Cloud
- ✓ Chapter 8. Multiple Processor Systems
- ✓ Chapter 9. Security
- ✓ Chapter 10. Case Study 1: UNIX, Linux, & Android
- ✓ Chapter 11. Case Study 2: Windows 8
- ✓ Chapter 13. Operating System Design
- ✓ Chapter 14. Reading List and Bibliography

