



香港中文大學

The Chinese University of Hong Kong

CSCI5550 Advanced File and Storage Systems

Lecture 00: Course Information

Ming-Chang YANG

mcyang@cse.cuhk.edu.hk

Course Information



- **CSCI5550 Advanced File and Storage Systems**
- **Course Time and Place**
 - **Lecture (*3)**
 - MON 09:30~10:15 (@ ERB 405)
 - TUE 13:30~15:15 (@ ERB LSB C4)
 - ~~**Tutorial (*1)**~~
 - ~~MON 10:30~11:15 (@ ERB 405)~~
- **PS. Check the course schedule for our arrangement.**
- **Course Website**
 - <http://www.cse.cuhk.edu.hk/~mcyang/csci5550/2020S/csci5550.html>

Course Instructor & Teaching Assistant

- **Course Instructor**

- Prof. Ming-Chang YANG (楊明昌)

- Office: SHB 906 (3943-8405)
- Office Hours: MON 14:30~16:30
- Email: mcyang@cse.cuhk.edu.hk



- **Teaching Assistant**

- Tsun-Yu YANG (楊尊宇)

- Office: *TBD*
- Office Hours: TUE 10:00~12:00
- Email: yangty@cse.cuhk.edu.hk

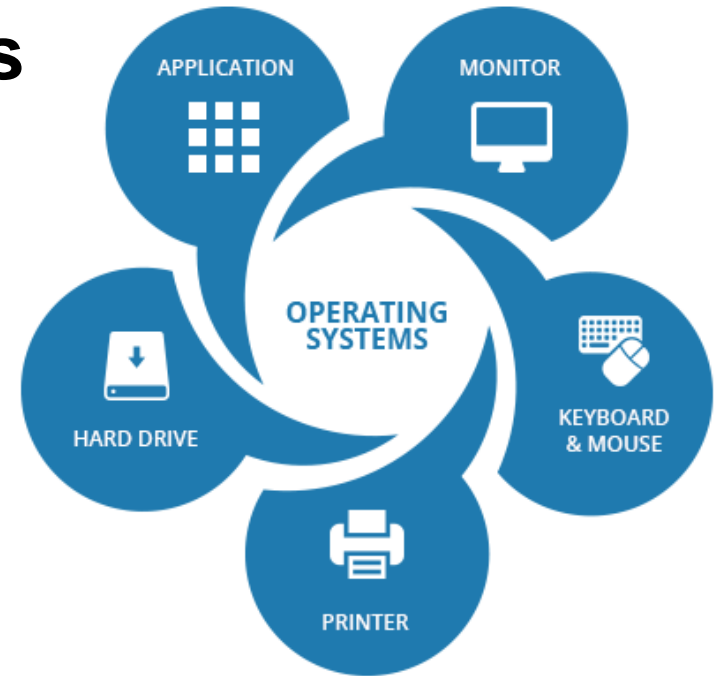


Pre-requisites



- **CSCI3150 Operating Systems**
(or equivalent)

- If you haven't taken it or have failed it, then you are advised **NOT** to take this course.
- Talk to me if you are uncertain.



- Comfortable with **C/C++ programming**



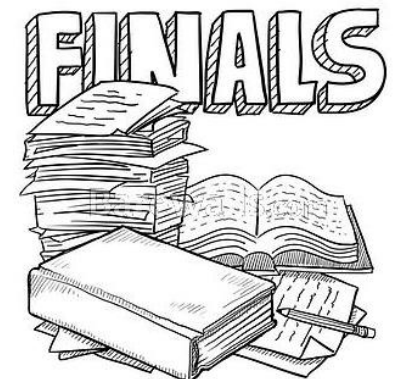
- Comfortable with **Linux**



Course Assessment



- **Programming Project (40%)**
 - Group of 1-2 students
- **Literature Survey & Presentation (30%)**
 - A literature survey report and presentation to prove your knowledge in file and storage systems
 - Group of 1-2 students
 - Peer grading
- **Final Exam (30%)**
- **Bonus (5%)**



Course Materials



- Suggested Readings

- Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau, **“Operating Systems: Three Easy Pieces”**

- Free online form: <http://pages.cs.wisc.edu/~remzi/OSTEP/>

Intro	Virtualization		Concurrency	Persistence	Appendices
<u>Preface</u>	3 <u>Dialogue</u>	12 <u>Dialogue</u>	25 <u>Dialogue</u>	35 <u>Dialogue</u>	<u>Dialogue</u>
<u>TOC</u>	4 <u>Processes</u>	13 <u>Address Spaces</u> <small>code</small>	26 <u>Concurrency and Threads</u> <small>code</small>	36 <u>I/O Devices</u>	<u>Virtual Machines</u>
1 <u>Dialogue</u>	5 <u>Process API</u> <small>code</small>	14 <u>Memory API</u>	27 <u>Thread API</u> <small>code</small>	37 <u>Hard Disk Drives</u>	<u>Dialogue</u>
2 <u>Introduction</u> <small>code</small>	6 <u>Direct Execution</u>	15 <u>Address Translation</u>	28 <u>Locks</u> <small>code</small>	38 <u>Redundant Disk Arrays (RAID)</u>	<u>Monitors</u>
	7 <u>CPU Scheduling</u>	16 <u>Segmentation</u>	29 <u>Locked Data Structures</u>	39 <u>Files and Directories</u>	<u>Dialogue</u>
	8 <u>Multi-level Feedback</u>	17 <u>Free Space Management</u>	30 <u>Condition Variables</u> <small>code</small>	40 <u>File System Implementation</u>	<u>Lab Tutorial</u>
	9 <u>Lottery Scheduling</u> <small>code</small>	18 <u>Introduction to Paging</u>	31 <u>Semaphores</u> <small>code</small>	41 <u>Fast File System (FFS)</u>	<u>Systems Labs</u>
	10 <u>Multi-CPU Scheduling</u>	19 <u>Translation Lookaside Buffers</u>	32 <u>Concurrency Bugs</u>	42 <u>FSCK and Journaling</u>	<u>xv6 Labs</u>
	11 <u>Summary</u>	20 <u>Advanced Page Tables</u>	33 <u>Event-based Concurrency</u>	43 <u>Log-structured File System (LFS)</u>	
		21 <u>Swapping: Mechanisms</u>	34 <u>Summary</u>	44 <u>Flash-based SSDs</u>	
		22 <u>Swapping: Policies</u>		45 <u>Data Integrity and Protection</u>	
		23 <u>Complete VM Systems</u>		46 <u>Summary</u>	
		24 <u>Summary</u>		47 <u>Dialogue</u>	
				48 <u>Distributed Systems</u>	
				49 <u>Network File System (NFS)</u>	
				50 <u>Andrew File System (AFS)</u>	
				51 <u>Summary</u>	

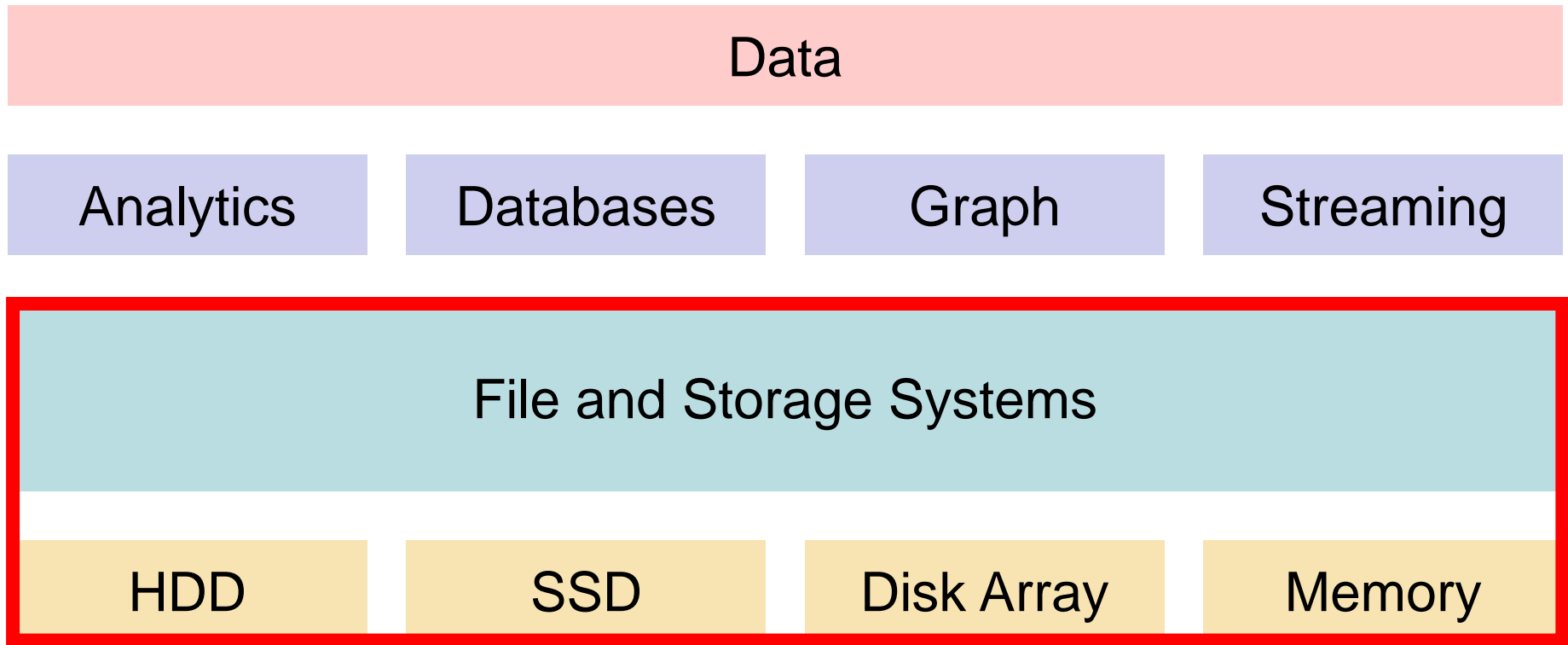
- **Papers** from top system conferences or highly referenced:

- E.g., FAST, OSDI/SOSP, USENIX ATC, EuroSys, NSDI, etc.

What We are Going to Learn (1/2)



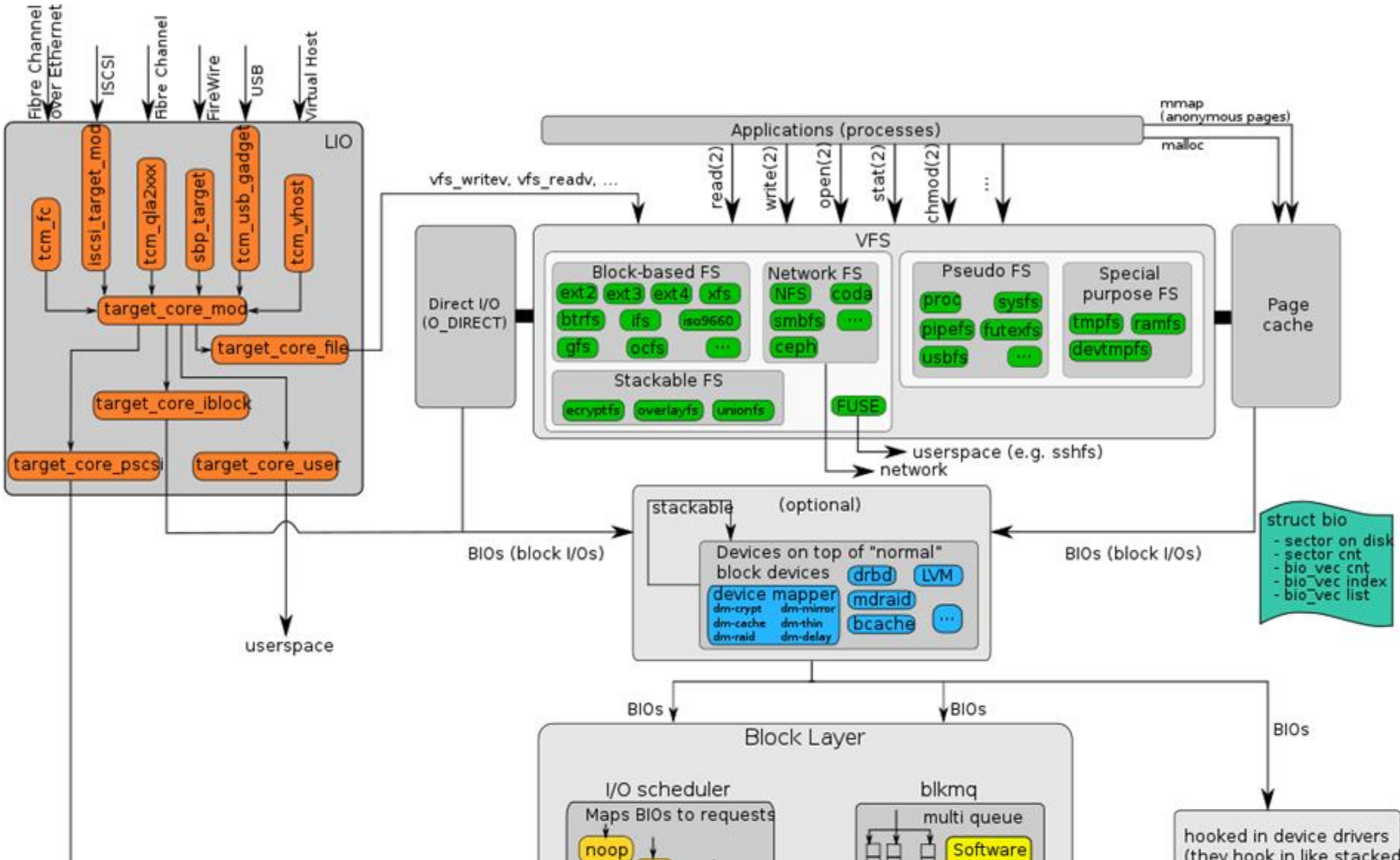
- Understand the *design* and *implementation* of **file and storage systems**, from a system perspective:



What We are Going to Learn (2/2)



- Understand the I/O stack of Linux kernel:



Make Your Own File System

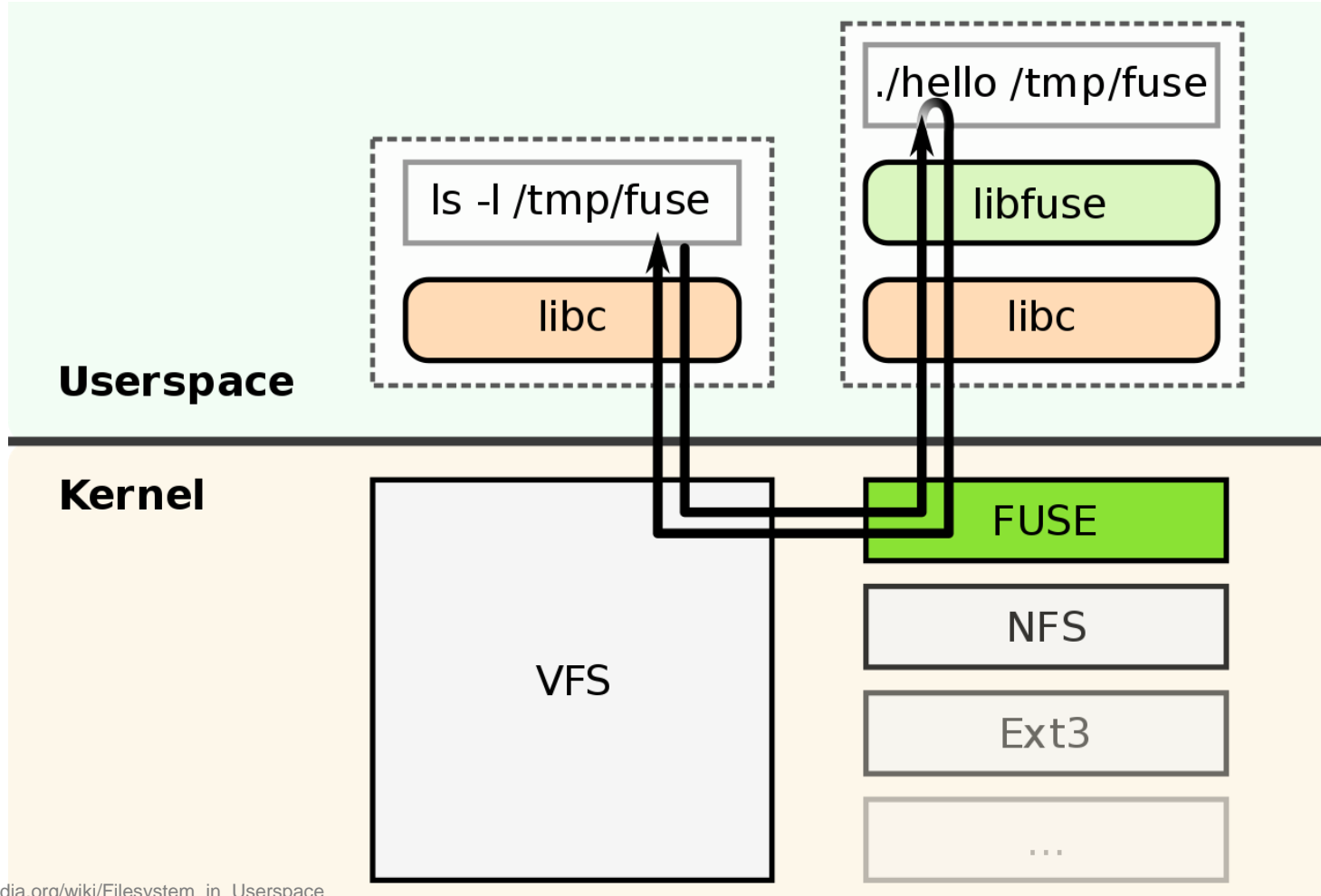


- **File System in Kernel-space**
 - Very **difficult** to build
 - Need careful use of **synchronization primitives**
 - **Only C language** supported
 - Standard C libraries **not available**
 - Need **root privilege**
- **File System in User-space (using FUSE!)**
 - Framework to implement user-space file system
 - User-space file systems trade **performance** for **flexibility**
 - **Easy to write**: Avoid awful coding in kernel
 - **Easy to test**: Run like a normal user program
 - **Easy to integrate libraries**: Can easily deploy libraries

FUSE: Filesystem in USErspace



- **FUSE** lets non-privileged users create their own file systems in user-space **without** editing kernel code.





- **First Phase: All-In-Memory File System**

- Implement an **all-in-memory file system** based on FUSE.
 - Both metadata (e.g., indexing structures of the file system) and file data are maintained in the memory.

- **Second Phase: In-Storage File System**

- Based on the first phase, convert the all-in-memory file system into an **in-storage file system**.
 - Both metadata and file data are persisted into a storage device (e.g., a USB flash drive).
 - The file data can still be accessed after re-mounting the user-space file system.

Course Schedule



	W	Date	Lecture	Suggested Readings	Tutorial
BASIC	1	Jan 6, 7	Lec01: I/O Devices	<i>OSTEP 36, 37</i>	-
	2	Jan 13, 14	Lec02: RAID and Data Integrity	<i>OSTEP 38, 45</i>	-
	3	Jan 20, 21	Lec03: File System Basics	<i>OSTEP 39~42</i>	-
	4	Jan 27, 28	Lunar New Year Vacation	No class	No tutorial
	5	Feb 3, 4	Lec04: File System Designs (I)	<i>OSTEP 43, 49, 50</i>	HW1
	6	Feb 10, 11	Lec05: File System Designs (II)	<i>GFS, Ceph</i>	-
	7	Feb 17, 18	Lec06: Benchmarking	<i>9Y-Study</i>	HW1 Q&A
ADVANCED	8	Feb 24, 25	Lec07: Solid-State Drives	<i>SSD-Tradeoffs, F2FS</i>	-
	9	Mar 2, 3	Lec08: New Hard Disk Drives	<i>SMR, IMR, SMR-FS</i>	-
	10	Mar 9, 10	Lec09: Memory Storage	<i>NVM, PMFS</i>	HW2
	11	Mar 16, 17	Lec10: Key-Value Storage	<i>LevelDB, RocksDB</i>	-
	12	Mar 23, 24	Lec11: Graph Storage	<i>GraphChi, BASC</i>	HW2 Q&A
	13	Mar 30, 31	Reading Week	No class	No tutorial
	14	Apr 6, 7	Literature Survey Presentation		Presentation
	15	Apr 13, 14	Easter / Presentation (Cont'd)		No tutorial

Important Notes



- Visit our course website regularly
- Plagiarism will **NOT** be tolerated
 - Don't copy!
 - Don't let other(s) copy!
 - Can discuss but write up the solutions by yourself!
- Honesty in Academic Work:
 - <http://www.cuhk.edu.hk/policy/academichonesty/>

The best way to learn is through practice!