CS444/544
Operating Systems II

Course Intro

09/24/20

Oregon State University

Instructor: Yeongjin Jang

- At OSU Since Oct 2017
- Main research area: systems security
- Hacking
  - CPU side-channel attacks, Jailbreaking, exploit development, automatic hacking (fuzzing / symbolic execution), designing secure systems, etc..
- Faculty advisor of OSUSEC
- CS419/579 Cyber Attacks and Defense
- CS419/579 Systems Security
- Feel free to reach me if you are interested in cybersecurity…
  - Join OSUSEC!

```
static void *
boot_alloc(uint32_t n)
{
    static char *nextfree;  // virtu
    char *result;

    // Initialize nextfree if this
    // 'end' is a magic symbol autor
    // which points to the end of tl
    // the first virtual address tha
    // to any kernel code or global
    if (!nextfree) {
        extern char end[];
        nextfree = ROUNDUP((char *)
    }
```

```
#define TRAPHANDLER_NOEC(name, num)                    \
    .globl name;                                       \
    .type name, @function;                             \
    .align 2;                                          \
    name:                                              \
    pushl $0;                                          \
    pushl $(num);                                      \
    jmp _alltraps
```

```
static inline physaddr_t
page2pa(struct PageInfo
{
    return (pp - pages)
}
```

```
void
env_pop_tf(struct Trapframe *tf)
{
    asm volatile(
        "\tmovl %0,%%esp\n"
        "\tpopal\n"
        "\tpopl %%es\n"
        "\tpopl %%ds\n"
        "\taddl $0x8,%%esp\n" /* skip tf_trapno and tf_errcode */
        "\tiret\n"
        : : "g" (tf) : "memory");
    panic("iret failed");  /* mostly to placate the compiler */
}
```

```
// These variables are set in
pde_t *kern_pgdir;      // Ker
struct PageInfo *pages;     //
static struct PageInfo *page_f
```

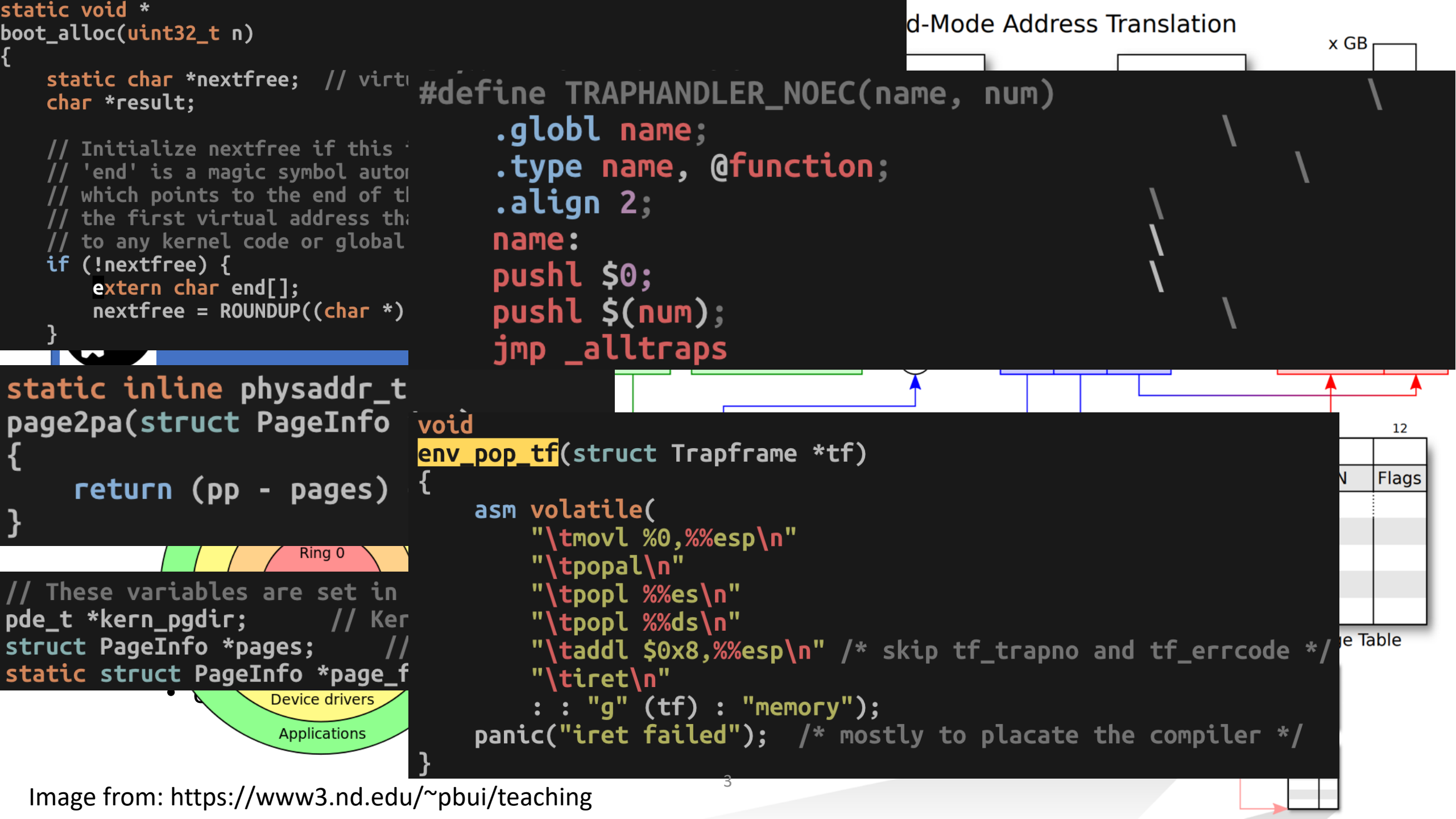


Image from: https://www3.nd.edu/~pbui/teaching

Course Objective

- Understand how modern computer systems work (in detail)
- Be able to answer the following questions:
  - What happens when we **turn on** the computer? How does it **boot**?
  - How an OS **runs an app**lication?
  - How an OS runs application that requires **memory more than its physical memory**?
  - How **multiple applications can run** on the system?
  - How an OS enforces **privilege separation**?
  - How an OS protects itself from **malicious software**?
  - How multiple programs **synchronize** each other? How can we **implement a lock**?

Important Links

- Website: https://os.unexploitable.systems
- Instructor: Dr. Yeongjin Jang (yeongjin.jang@oregonstate.edu)
- TAs:
  - Hadi Rahal-Arabi, Andrew Quach, and Phillip Mestas III (graduate TAs)
  - Laura Jiang, Lyell Read, and Corbin Tegner (undergraduate TAs)

- Gitlab: https://gitlab.unexploitable.systems
- Piazza: https://piazza.com/class/ket6bdq5r6y55
- Discord: https://discord.gg/hPyPaQf
- Assignment server: os2.engr.oregonstate.edu, os1, oldos1, oldos2

Course Structure

- 10 weeks schedule
  - https://os.unexploitable.systems/cal.html
  - Virtualization (Week 1-4)
  - Concurrency (Week 5-8)
  - Persistency and others (Week 9-10)
- Textbook
  - http://pages.cs.wisc.edu/~remzi/OSTEP/

Monday	Tuesday	Wednesday	Thursday	Friday
Sep 21	Sep 22	Sep 23	Sep 24 **LEC 1:** Intro to the course **Watch 1:** Lecture #1 VIDEO PDF PPTX **Watch 2:** Lab tutorial 1 VIDEO PDF PPTX **Study, Lab 1:** Booting a PC **Read:** Textbook **Read:** at&t_asm GDB tutorial1 tutorial2 cheat-sheet **Read:** tmux cheatsheet (ctrl-b -> backtick) tmux-cheat-sheet *First day of class*	Sep 25
Sep 28	Sep 29 **LEC 2:** BIOS/Booting/CPU	Sep 30	Oct 1 **LEC 3:** Memory: Address Space, Segmentation, and Paging **Read:** x86_Address_Translation **Read:** Textbook1 Textbook2 Textbook3	Oct 2
Oct 5 **DUE:** Lab 1 (100%)	Oct 6 **LEC 4:** Virtual Address Translation **Study, Lab 2:** Memory Management **Read:** Page_Table	Oct 7	Oct 8 **LEC 5:** Virtual Memory Layout **Read:** Textbook1 Textbook2 Textbook3 Textbook4 Textbook5 Textbook6	Oct 9
Oct 12 **DUE:** Lab 1 (75%)	Oct 13 **LEC 6:** JOS Memory Management	Oct 14	Oct 15 Quiz 1: Virtual Memory	Oct 16

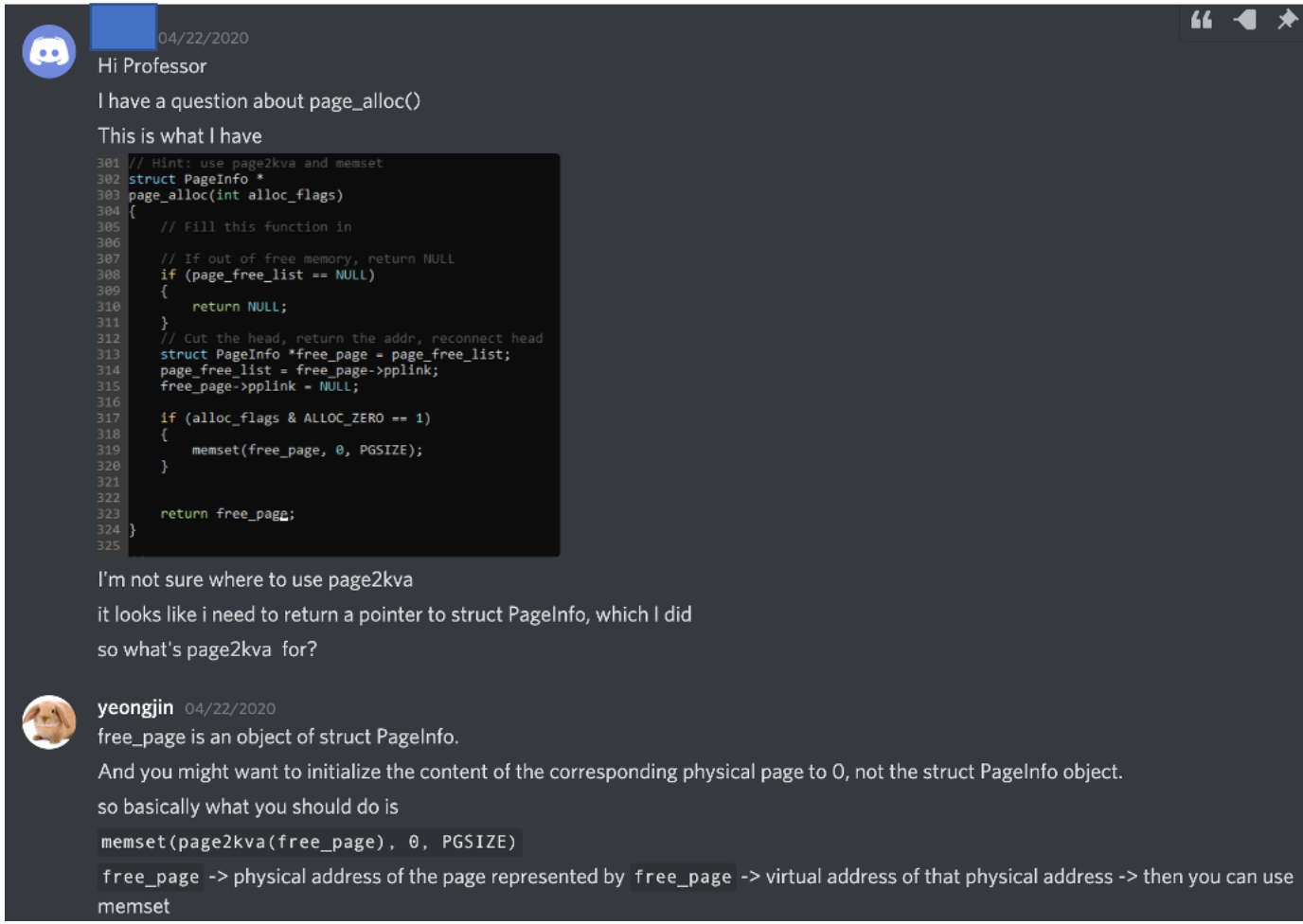
- **Read**: prep materials posted on homepage
- **Watch**: 50 min VIDEO lecture
- **Study**: study JOS labs (tutorial videos / lab instructions website)
- **Engage**: ~30 min ZOOM Q&A and office hours on Discord, discuss with peers!

Meeting Time (with me)

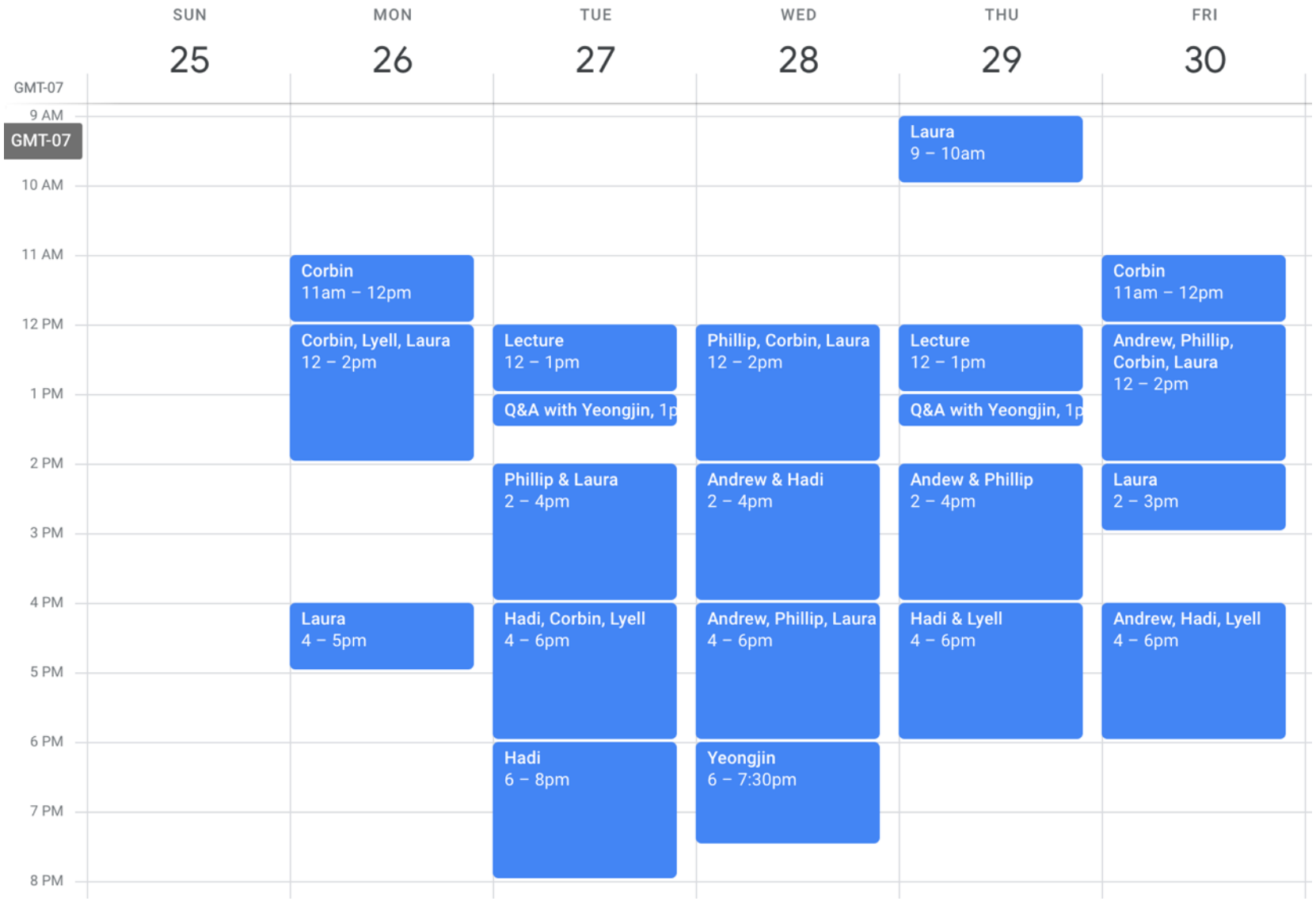
- Lecture (Video, **asynchronous**)
  - Released at Tu/Th 12:00pm
  - Video, link will be available on Canvas/Homepage
- Lecture Q&A (Synchronous via Zoom, not required)
  - Tu/Th 01:00pm ~ 01:30pm (Zoom link on Canvas)
- My office hour (via Discord)
  - Wed 06:00pm – 07:30pm (https://discord.gg/hPyPaQf)

Recitation / TA Office Hours

- Available via Discord
  - https://discord.gg/hPyPaQf
- We will use recitation section times as TA office hour as well
  - Total 28.5 hours of office hours
  - Monday: 4 hours
  - Tuesday: 6 hours
  - Wednesday: 7.5 hours
  - Thursday: 5 hours
  - Friday: 6 hours



Office Hours, etc.



https://calendar.google.com/calendar/u/0/embed?src=06a8uv6u8f01gkg4nljsc35kqg@group.calendar.google.com&ctz=America/Los_Angeles&mode=WEEK

Move to the week of 9/25~10/2

Grading

- 70% JOS lab assignment
  - Lab 1 (10%), Lab 2 (15%), Lab 3 (20%), Lab 4 (25%)
- 30% Quizzes (mini-exam)
  - Quiz 1 (10/15) : Virtual Memory
  - Quiz 2 (11/05): System calls, faults, and exceptions
  - Quiz 3 (11/19): Concurrency

Grading Scheme (tentative):

100 >= A >= 93 (96 for graduate students)

93 > A- >= 90 (93)

90 > B+ >= 86 (89)

86 > B >= 83 (86)

83 > B- >= 80 (83)

80 > C+ >= 76 (79)

76 > C >= 73 (76)

73 > C- >= 70 (73)

70 > D+ >= 66 (69)

66 > D >= 63 (66)

63 > D >= 60 (63)

F < 60 (63)

The Lab (70%)

- Four labs
  - JOS Lab 1 (10%): Booting a PC (1.5 weeks, due on 10/5)
    - Bootloader, protected mode, etc.
  - JOS Lab 2 (15%): Memory Management (2 weeks, due on 10/19)
    - Virtual memory, paging, etc.
  - JOS Lab 3 (20%): User Environment (3 weeks, due on 11/9)
    - Process, user, kernel, system call, etc.
  - JOS Lab 4 (25%): Preemptive Multitasking (3.5 weeks, due on 12/4)
    - Implementing context switching, multi-core support, inter-process communication, etc.

How to Conduct Lab Assignments?

- Visit Lab Tutorial Webpage
  - https://os.unexploitable.systems/lab/lab1.html
- Watch Lab Tutorial Video
  - I will explain necessary stuff for the lab assignments in the video (code/examples, etc.) and also share some tips…

An Exercise Example in Lab 1

Note

Exercise 3. Take a look at the lab tools guide, especially the section on GDB commands. Even if you're familiar with GDB, this includes some esoteric GDB commands that are useful for OS work.

Set a breakpoint at address 0x7c00, which is where the boot sector will be loaded. Continue execution until that breakpoint. Trace through the code in `boot/boot.S`, using the source code and the disassembly file `obj/boot/boot.asm` to keep track of where you are. Also use the *x/i* command in GDB to disassemble sequences of instructions in the boot loader, and compare the original boot loader source code with both the disassembly in `obj/boot/boot.asm` and GDB.

Trace into `bootmain()` in `boot/main.c`, and then into `readsect()`. Identify the exact assembly instructions that correspond to each of the statements in `readsect()`. Trace through the rest of `readsect()` and back out into `bootmain()`, and identify the begin and end of the `for` loop that reads the remaining sectors of the kernel from the disk. Find out what code will run when the loop is finished, set a breakpoint there, and continue to that breakpoint. Then step through the remainder of the boot loader.

The Lab Could be

- Coding KERNEL code in C
  - Any memory error -> Triple fault
- Us
- As
- Co
  - Page table
  - Global descriptor table (GDT)
  - Interrupt descriptor table (IDT)

Watch both lectures and lab tutorial videos on time and ask TAs for help!

Intel® 64 and IA-32 Architectures
nual
& 3D):
Guide

		IA-32e	CR3	47:39	N/A (PS must be 0)
Page-directory-pointer table	PDPTE	32-bit	N/A		
		PAE	CR3	31:30	N/A (PS must be 0)
		IA-32e	PML4E	38:30	1-GByte page if PS=1[1]
Page directory	PDE	32-bit	CR3	31:22	4-MByte page if PS=1[2]
		PAE, IA-32e	PDPTE	29:21	2-MByte page if PS=1
Page table	PTE	32-bit	PDE	21:12	4-KByte page
		PAE, IA-32e		20:12	4-KByte page

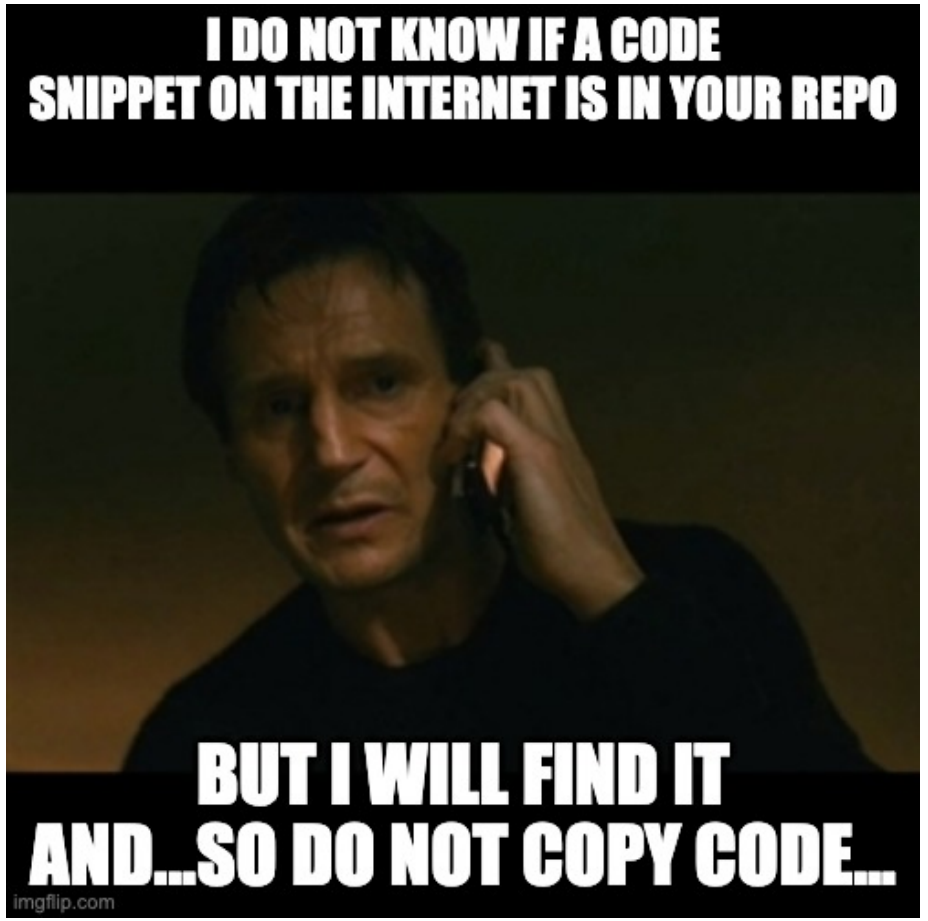
Lab Rules

- DO NOT SHARE YOUR CODE WITH OTHER STUDENTS
  - **You are encouraged to discuss with others** about the assignments but do not ask/give the code to the others
  - **Do not copy** other students' code or code available in online
  - **Do not publish** your code online

- You will be asked to submit a simple write-up for the assignment
  - Describe how you solve each exercise/questions
  - Mention your collaborators in the write-up
  - **Do not copy** other students' write-up
  - **Do not publish** your write-up online

Lab Rules

- Plagiarism will be punished via the Office of Student Life..
  - E.g., getting F or zero point for the lab assignment that matters with plagiarism…
- Please refer the Code of Student Conduct
  - https://studentlife.oregonstate.edu/studentconduct/academicmisconduct
  - https://studentlife.oregonstate.edu/sites/studentlife.oregonstate.edu/files/edited_code_of_student_conduct.pdf



Lab Rules – Late Submissions

- If you submit your assignment before the due date, then
  - You will get 100% of credit from ‘make grade’ of your submission
- If you submit your assignment within one week after the due date, then
  - You will get 75% of credit from ‘make grade’ of your submission
- If you submit your assignment before 12/11 11:59 pm, then
  - You will get 50% of credit from ‘make grade’

Due Dates on the Calendar

Oct 5 **DUE:** Lab 1 (100%)	Oct 6 **LEC 4:** Virtual Address Translation **Study, Lab 2:** Memory Management **Read:** Page_Table	Oct 7	Oct 8 **LEC 5:** Virtual Memory Layout **Read:** Textbook1 Textbook2 Textbook3 Textbook4 Textbook5 Textbook6	Oct 9
Oct 12 **DUE:** Lab 1 (75%)	Oct 13 **LEC 6:** JOS Memory Management	Oct 14	Oct 15 Quiz 1: Virtual Memory	Oct 16
Oct 19 **DUE:** Lab 2 (100%)	Oct 20 **LEC 7:** Quiz 1 Review **Study, Lab 3:** User Environment	Oct 21	Oct 22 **LEC 8:** User/Kernel Switch **Read:** Textbook-process Textbook-syscall Textbook-trap	Oct 23
Oct 26 **DUE:** Lab 2 (75%)	Oct 27 **LEC 9:** Handling Interrupts/Exceptions	Oct 28	Oct 29 **LEC 10:** System Calls and Page Fault	Oct 30
Nov 2	Nov 3 **LEC 11:** Virtualization Recap and Quiz 2 Prep	Nov 4	Nov 5 Quiz 2: System calls, faults, and exceptions	Nov 6
Nov 9 **DUE:** Lab 3 (100%)	Nov 10 **LEC 12:** Multi-threading and Synchronization **Study, Lab 4:** Preemptive Multitasking **Read:** READ Concurrency Thread	Nov 11	Nov 12 **LEC 13:** Lock and Thread Synchronization	Nov 13
Nov 16 **DUE:** Lab 3 (75%)	Nov 17 **LEC 14:** Concurrency Bugs and Deadlock **Read:** READ Bugs	Nov 18	Nov 19 Quiz 3: Concurrency	Nov 20
Nov 23 Thanksgiving break	Nov 24 Thanksgiving break	Nov 25 Thanksgiving break	Nov 26 Thanksgiving break	Nov 27 Thanksgiving break
Nov 30	Dec 1 **LEC 15:** TBD	Dec 2	Dec 3 **LEC 16:** Final Summary *The last day of class*	Dec 4 **DUE:** Lab 4 (100%)
Dec 7 No Final Exam!	Dec 8 No Final Exam!	Dec 9 No Final Exam!	Dec 10 No Final Exam!	Dec 11 **DUE:** Lab 1,2,3 (50%), 4 (75%)

CS 544 Students

- Will have higher grade bar than CS444 (+3pts)
  - E.g., A is for 93 and over for CS444, and 96 and over for CS544

Grading Scheme (tentative):

100 >= A >= 93 (96 for graduate students)

93 > A- >= 90 (93)

90 > B+ >= 86 (89)

86 > B >= 83 (86)

83 > B- >= 80 (83)

80 > C+ >= 76 (79)

76 > C >= 73 (76)

73 > C- >= 70 (73)

70 > D+ >= 66 (69)

66 > D >= 63 (66)

63 > D >= 60 (63)

F < 60 (63)

Tips to the Lab

- Study in a group (discussions are highly encouraged!)
  - But please **write the code individually**!
- Follow tutorial video
- Ask questions (Piazza, Discord)
- Understand your time budget (debugging will take **lots of your time**!)
  - **Plan ahead** to finish the labs on time
- Learn basic tools (e.g., C, gdb, assembly, editors, tmux, etc.) ASAP
  - This will help you earn more time on doing labs…
  - https://missing.csail.mit.edu/
  - Up to Debugging and Profiling would be helpful…

Notices

- There will be lecture Q&A from 1:00 pm to 1:30 pm on every Tuesday and Thursday

- My (virtual) office hour is Wed 6:00 pm ~ 7:30 pm (Discord)