The second exam will be in class on Monday, March 30. If you have an unavoidable conflict with the date of an exam, please see me as soon as possible (before the exam date!) to discuss options for rescheduling. Last minute rescheduling/extensions will not be accommodated for something known about in advance.

The exam covers material from OSTEP chapters 9, 12–16, and 18–20 (fair share scheduling and virtual memory up to but not including swapping).

The exam is closed book. You may have a single page of notes (8.5x11", one side) which will be handed in with your exam. This page may be handwritten or typed and can contain whatever you would like, but it must be a hardcopy — on a piece of paper, not a laptop, tablet, phone, or other device — and must be personally prepared by you — you may not copy another student's page or hand out copies of yours to others. Creating your own notes is an essential part of the learning process — deciding what to include requires engagement with the material which reinforces understanding and improves long-term retention of the material, provides an opportunity for review in order to identify gaps in your knowledge in time to ask questions before the exam, increases confidence in what you do know, and encourages taking ownership of your own learning.

Terms and ideas that you should be familiar with:

- lottery scheduling
  - terms and concepts: proportional-share scheduling, fair-share scheduling, tickets
  - ticket mechanisms: ticket currency, ticket transfer, ticket inflation
    - what each is
    - why/how each can be used
  - stride scheduling
    - what it is / how it works
    - compare and contrast with lottery scheduling
  - advantages, drawbacks, limitations of lottery scheduling and stride scheduling
- memory virtualization
  - terms and concepts: address space, virtual address, physical address, sparse address space
  - address space layout
- memory API and memory management
  - system calls: `malloc`, `free`

  - terms and concepts: dynamic memory allocation, memory leaks, dangling pointers, internal vs external fragmentation
  - goals of virtual memory systems
- address translation
  - terms and concepts: address translation
  - mechanisms: base and bounds (dynamic relocation)
  - hardware support for address translation: base and bounds registers, MMU (memory management unit)
  - virtual-to-physical mapping
- segmentation
  - concepts and terms: segments, segmentation fault / segfault
  - logical divisions of memory: code, stack, heap and their roles when a process is running
  - hardware support for segmentation: base and bounds registers, direction of growth bit
  - OS support for segmentation
  - address translation with segmentation (segment number, offset)
  - benefits and challenges of segmentation
- paging
  - concepts and terms: paging, pages, page frames, page table, page table entry (PTE), virtual page number (VPN), offset, physical frame number (PFN), page fault, trap handler
  - hardware support for paging: page table base register (PTBR), valid bit, protection bits
  - address translation with a linear page table
  - benefits and challenges of paging
- TLBs
  - concepts and terms: translation lookaside buffer (TLB), caching, TLB hit, TLB miss, TLB hit rate, locality, spatial locality, temporal locality
  - address translation with TLB lookup
  - hardware- and software-managed TLBs
    - what each is / how each works
    - handling TLB misses and page faults
    - pros and cons of each approach
  - TLB replacement policies
  - handling context switches
  - what issue(s) TLBs address, benefits and challenges of TLBs
- other page table organizations
  - the problem being addressed

  - multi-level page tables
    - what they are
    - how address translation works
    - benefits and challenges
  - inverted page tables
    - what they are
    - how address translation works
    - benefits and challenges

Things you should be able to do include:

- lottery and stride scheduling
  - describe, compare, and contrast lottery scheduling and stride scheduling
  - compute CPU share given ticket counts
  - explain how ticket transfer or inflation changes CPU share
  - simulate different lottery draws and determine which job runs when
  - simulate stride scheduling
  - compute the fairness of scheduling policies
  - compute scheduling order using stride scheduling
- memory virtualization
  - identify the components of a typical address space layout and how they are arranged
  - explain why virtualization is necessary
- address translation
  - base and bounds address translation: given base and bounds registers and a virtual address, determine whether the address is valid and, if so, compute the physical address
  - segmentation address translation: given segment base and limit registers and a virtual address, determine whether the address is valid and, if so, compute the physical address
  - paging: given the page size, page table, and a virtual address, determine the VPN and offset, look up page table entries, perform virtual-to-physical translation, and determine whether a page fault occurs
  - TLBs: trace address translation with a TLB lookup, determine whether an access is a TLB hit or miss, explain what happens on a TLB miss, and explain TLB flushing on context switches
  - describe how TLBs speed up translation
- paging and advanced page tables

  - given the virtual address size, page size, and page table entry size, compute the size of the page table
  - explain why page tables can become large and describe methods for reducing page table size
  - perform address translation with multi-level page tables
  - explain how inverted page tables work

- compare and contrast segmentation and paging

You should also be able to do computational problems of the types assigned in homeworks 3-5 (from OSTEP chapters 9, 15–16, 18, 20).