☑ 1. Last time
☑ 2. Final exam
☑ 3. Your questions
☑ 4. Wrap-up

---

2. Final exam

- 110 minute exam
  - ~~stay seated at 100 mins~~
  - closed book
  - TWO two-sided sheets allowed

Material

- Readings
- Labs
- HWs
- Classes

[see midterm topic list] → see l12.txt

Post-midterm topics (not guaranteed to be necessary or sufficient)

virtual memory

paging

virtual memory on x86-64

virtual address [0000| 36 bits ... |12 bits]

virtual

entry in L1...L4 page tables

[40 bits more bits bottom 3 bits]

protection (u/s | R/W | NP/P)

what's a TLB?

page faults

mechanics

costs

uses

page replacement policies (FIFO, LRU, CLOCK, OPT)

thrashing

mmap()

I/O

architecture

how CPUs and devices interact

mechanics

polling vs. interrupts

DMA vs. programmed I/O

device drivers

synchronous vs. async I/O

context switches

User-level threading

Disks

- geometry
- performance
- interface
- scheduling (skipped in class, covered in book)

File systems

- basic objects: files, directories, metadata, links, inodes
- how does naming work?
- types of file layout
    - extents/contiguous, linked, index
    - classic Unix + FFS are variants of indexed
- analogy between inode and top-level page directory (aka L1 page table)
- tradeoffs
- performance

Crash recovery

- ad hoc
- copy-on-write (cow)
- journaling (redo logging, undo logging, undo+redo)
  WAL

RPC, client/server systems

case study: NFS

marquee user of RPC

RPC: transparent or not?

protection and security

stack smashing / buffer overflow

Unix security model

access control, privileges, setuid, attacks

trusting trust

bootup, from power-on

static linking + loading is a key tool

bootstrap process

H/W copies firmware into read/write mem

firmware is mini OS

runs bootloader program, which ultimately begins kernel

kernel invokes init(1) / init(8)

init(1) invokes login(1)

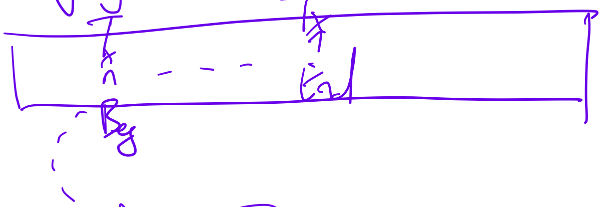
login(1) lets you get a shell and begin executing programs

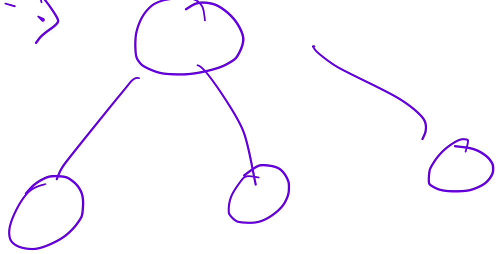
Redo + undo logging

Redo:

Txn Beg

log/journal





Txn End

Redo + Undo:

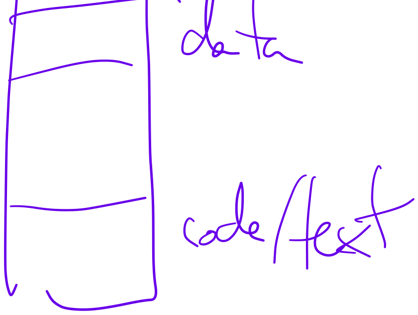
10, 22, 20, 2, 40, 6, 38   6 msec

↓
20

20, 22, 10, 6, 2, 38, 40

2   12   4   4   36   2   360

W ~ X


data
code/text