Final Exam Review 12/12

Final exam Logistics

Exam : 12pm-1:45pm Dec 17 in this calssroom

If you finish before 105 mins, you can submit early up to the 95 mins.

Must hand-in your exam to me, show me your ID.

I will leave the classroom 1:50pm

Cheatsheet: 1 letter size page, double-sided, print/write whatever you want

Please submit the cheatsheet with your exam, Please write your name on the cheatsheet

Topics covered: everything (more focused on the slides, and programming, and homework)

Warning:

- Please write the answer in a way I can understand
- If there is any ambiguity you find in the question, write down your assumption.

Topics we covered in class:

- high level overview of operating system, what is a OS and why it exists
- Processes: process'es view of memory and registers
    - stack frames
    - OS views of processes
- system calls (exec/fork)
- Process/OS passing controls
- Shell (lab 2), file description, redirections, pipes
- Concurrency, threading, ...
    - critical sections
    - how to make sure things are atomic
    - no race conditions
    - mutexes, spinlocks, condition variables, monitors (lab 3)
    - what things can go wrong
        - deadlock (liveness), safety (correctness)

  - performance, made tradeoff w.r.t complexity, correctness
  - linearizability
- software safety (therac-25)
- scheduling (when scheduling happens, what metirc, what costs), specifics algorithms
- (after midterm) context swtiching
- memory mangement: virtual memeory (lab 4)
  - paging, page tables, multi-level page tables, protections, TLB,
  - page faults (mechanism, costs and uses)
  - page replacement policies (FIFO, LRU, CLOCK, Optimal)
  - Thrashing
  - This is high relevant to the concept caching
- [Everything up to midterm]
- I/O (architecture, how CPUs and devices interact: mechanics, polling vs interrupts)
  - device drivers
- Disks (geomtry, performance, interface)
- File systems (lab 5)
  - basic object: files, directories, meta-data, links, inodes
  - how naming works
  - file layout (contiguous, linked, indexed sturcture)
    - Unix and FFS are variants of indexed structure
    - tradeoff/performance
  - Consistency? Crash recovery: ad-hoc, copy-on-write, journalling (redo-logging, undo-logging)
- Protection and security
  - stack smashing / buffer overflow
  - Trusting trust
  - Unix security (access control, privileges, setuid, attacks)
- The process from booting the computer to getting the terminal

Lab 4:

```
make grade-N
```

Lab 5: grading script available