Final Exam Review

CS 450: Operating Systems
Sean Wallace <swallac6@iit.edu>
Main Topics

- Synchronization & Concurrency
- Building Concurrency Primitives
- Deadlock
- Filesystems & RAID
- xv6
Synchronization Topics

- Why? (How do race conditions come to be?)
- TSL and semaphore semantics / usage
- Basic criteria for concurrent programs (no starvation, bounded waiting, etc.)
- Essential synchronization patterns: rendezvous, turnstile, barrier, light-switch
Concurrency Topics

• Disadvantages of shared memory models
  • Complexity of locking
  • Race conditions, deadlock, lowered throughput
• Use of semaphores for synchronization
  • 4 rules, essential criteria, etc.
Building Concurrency
Primitives Topics

- Lock implementation
- Spinlocks and their disadvantages
- Test/set (atomic exchange)
- Sleep/wakeup
Deadlock Topics

• Necessary conditions for deadlock

• Resource allocation graphs

• Deadlock avoidance: algorithm (Banker’s) and require assumptions

• Deadlock detection: algorithm

• Possible approaches to recovery
Filesstem Topics

• Essential goals - device agnosticism: desirable/achievable?

• Allocation mechanisms (contiguous/extents, FAT, i-node)

• Free space tracking (free space bitmap)

• Hardware: HDDs and SSDs
Filesystem Topics

• ACID properties and FS/user responsibilities
  • Crashes & possible issues
    • Soft updates: corruption vs. lost blocks
  • Logging for atomicity/consistency (journaling)
• RAID techniques and levels: pros/cons (very high level)
• No coding

• Similar to midterm, high level idea of what’s going on (if you did the assignments, you already know what you need to know)