CSI 450 — Operating Systems, FALL 2009

Review Sheet #1

- Chapter 1

    - Single processor vs. multiprocessor
    - cluster systems
    - OS structure (multiprogramming, time-sharing)
    - processes, job pool, job scheduling, CPU scheduling, swapping, virtual memory
    - Interrupts (hardware and software)
    - kernel mode vs. user mode (privileged instructions)
    - timers and other resources
    - OS responsibilities (processes, files, caching etc)
    - protection and security
    - distributed systems and real-time systems

- Chapter 2

    - OS services for the user
    - OS services for the system
    - UI vs GUI
    - System calls
    - System call interface and groups
    - System programs
    - Mechanism vs. Policy
    - OS languages (high level vs low level)
    - OS Structure (simple,layer, microkernel, modules)
    - Virtual machines
    - system boot (bootstrap, implementation)

- Chapter 3

    - Processes (user vs. OS)
    - processes in memory
    - program vs. process
    - The states of a process

- Process Control Block
- Process Scheduling (scheduler, job queue, ready queue, queuing diagram)
- Long-term vs. short-term scheduler
- I/O bound vs. CPU bound processes
- context-switch
- process tree
- Resources of a child process (advantages and disadvantages)
- Concurrent execution of process (or wait)
- System calls (fork, execlp, wait, perror,exit)
- process creation diagram
- Process Termination
- Early termination of the child (advantages)
- Independent vs. Cooperating processes
- Reasons to allow process to share (information sharing, computational speedup, modularity, convenience)
- IPC mechanisms (two fundamental models, diagrams)
- Shared memory
  * Advantages
  * Implementation
  * Synchronization
  * Unbounded vs. bounded buffer
  * System calls (shmget, shmat, shmdt, shmctl)
- Message Passing Systems
  * direct communication (properties, symmetry and asymmetry)
  * indirect communication (properties)
  * Ownership of a mailbox (process, OS, both)
  * Synchronous (blocking) vs. Asynchronous (non-blocking)
  * buffers (zero capacity, bounded, unbounded)