MAT 450 — Operating Systems, Spring 2005

Review Sheet #2

- Chapter 4 - Threads
  - heavyweight vs. lightweight (p127)
  - The register/stack/code/data/files view of threads of execution (p127)
  - Four major categories of benefits of multiprogramming - responsiveness, resource sharing, economy, multiprocessor utilization (p129)
  - user threads vs kernel threads (p129)
  - many-to-one, one-to-one, many-to-many (p130)
  - pthreads (POSIX THREADS) - thread library - API (p132-133)
  - pthread_attr_init, pthread_create, pthread_join, pthread_exit
  - Threading issues (fork?, exec?) (p138)
  - thread cancellation, target thread, asynchronous, deferred, cancellation points (p139)
  - signal handling, signal handling process, synchronous, asynchronous (p139)
  - default signal handler vs. user-defined signal handler (p140)
  - delivery (thread who signaled, every thread, certain threads, a specific thread) (p140)
  - thread pools (e.g. web server) (p141)
- Chapter 5 - CPU Scheduling
  - multiprogramming goal (p153)
  - CPU burst, I/O burst, cycle of processes (p154)
  - large # of short CPU bursts (p155)
  - I/O bound vs CPU bound (p155)
  - Short-term scheduler selects a process when a process switches from: running to waiting, running to ready, waiting to ready, running to terminated (p156)
  - nonpreemptive (cooperating) schedule scheme vs. preemptive scheduling scheme (p156)
  - dispatcher (switch context, switch to user mode, jump to proper location in user program), dispatch latency (p157)
  - scheduling criteria (CPU utilization, throughput, turnaround time, waiting time, response time)(p157)

  - Scheduling Algorithms, Gantt chart, FCFS, SJF, priority scheduling (internal or external), RR, burst prediction, time quantum, context switch delay (p158–166)

- Chapter 6 - Process Synchronization
  - producer/consumer, bonded buffer, concurrent execution (p191,192)
  - The critical section problem (p193)
  - The critical section solution (mutual exclusion, progress, bounded waiting) (p194)
  - race conditions, preemptive kernel vs nonpreemptive kernel (p194)
  - Peterson's Solution (p195)
  - atomic instructions,TestAndSet, Swap (p197–199)
  - semaphores, counting, binary, mutex, synchronization (p201)
  - implementation- busy waiting, spinlock, or blocking (p202)
  - deadlock and starvation (p204)
  - classic problems - bounded buffer, dining philosophers, sleeping barber (p205–209)