Final Study Guide

This is simply a guide of topics that I consider important for the final. I don't promise to ask you about them all, or about any of these in particular; but I may very well ask you about any of these, as well as anything we discussed in class or that is in the reading.

1. Anything from the midterm study guide
2. Input and Output
    a. device drivers and transparency
    b. structure of a device driver
    c. character code independence, device independence, uniform treatment of devices
    d. escape characters, bit stuffing
    e. device interfaces
    f. device drivers: lower, upper parts
    g. disk scheduling algorithms: FCFS, pick-up, SSTF, SCAN, LOOK, N-Step SCAN, C-SCAN, C-LOOK
    h. file, system calls for I/O
    i. blocking vs. non-blocking I/O
3. Memory management
    a. bare machine, resident monitor, fence addresses, and fence registers
    b. relocation and address binding
    c. bounds registers, base and limit registers
    d. internal vs. external fragmentation
    e. compaction
    f. paging and page tables
    g. segmentation and segment tables
    h. optimizations: cache, hit ratio, effective memory access time
    i. views of memory: program vs. operating system, address translation
    j. protection
    k. segmented paging (segment the page table)
    l. paged segmentation (page the segments)
    m. virtual memory: demand paging, page faults, pure demand paging
    n. page replacement and victims and dirty bits: FIFO, OPT, LRU, stack algorithms
    o. minimum number of pages per process
    p. global vs. local allocation
    q. working set: thrashing, principle of locality, working set model: working set, window size, working set principle
    r. prepaging, I/O interlock, choosing page size, restructuring program
4. File Systems
    a. virtual vs. physical; names; directory structures
    b. access control: rights, ACLs, UNIX abbreviations
    c. access via create, open, close, read, write, rewind, delete system calls or commands
    d. access methods: sequential, direct mapped, structured
    e. disk directory: free list implementations, allocation methods (contiguous, linked, indexed)
5. Deadlock
    a. resource manager, request, release
    b. what is deadlock; difference between it and starvation
    c. resource types: reusable, consumable
    d. how to deal with deadlock: ignore, detection and recovery, prevention (mutual exclusion, no preemption, circular wait, hold and wait), avoidance
    e. deadlock recovery: breaking circular wait, break no preemption (i.e., allow preemption)
    f. deadlock prevention: hierarchical ordering (ordered resource) policy, acquire all resources before running
    g. deadlock avoidance: banker's algorithm
6. Computer Security