Outline for January 18, 2001

1. Greetings and felicitations!
   a. All projects turned in are on the web page; you should have received approval or disapproval by now
2. System model
   a. Process maps one state into a set of states (each a potential ending state)
   b. Define blocked, deadlocked process; deadlocked, safe states
   c. Resource graphs; request, assignment edges; operations are requesting, acquiring, releasing
   d. Review terms: bipartite, sink, isolated nodes, path, cycle, reachable set, knot
3. Deadlock Detection
   a. Graph analysis of system: assume serially reusable resources (SRR)
   b. Reduction of SRR graphs
   c. Lemma: All reduction sequences of a given SRR graph lead to the same irreducible graph
   d. Deadlock Theorem: *S* is a deadlock state if and only if the reusable resource graph of *S* is not completely reducible.
   e. Cycle Theorem: A cycle in a reusable resource graph is a necessary condition for deadlock.
   f. Continuous deadlock detection
   g. Expediency and deadlocks
   h. Single-unit resources and deadlocks
4. Deadlock Recovery
   a. Process termination: kill one with lowest cost first
   b. Termination in expedient states, single unit requests: terminate one process per knot, minimum cost to restart
   c. Process pre-emption
5. Deadlock Prevention
   a. Requirements for deadlock: mutual exclusion, hold and wait, no pre-emption, circular wait
   b. Collective request policy
   c. Pre-emption
   d. Ordered request policy
6. Deadlock Avoidance
   a. Prevent system from ever entering an unsafe state
   b. Maximum claim graph
   c. Example: Banker's algorithm
7. Consumable Resources
   a. What are they
   b. General properties
   c. Special cases
8. Known Producers, Unknown Consumers
   a. Resource graph
   b. Deadlock detection and recovery
9. Known Producers, Known Consumers
   a. Claim-limited state
10. General Resource Graph