CSE/EE 461 Winter 2004 Final Exam Format

The format will be roughly as follows:

Type	Number of questions	Points each
Multiple choice	10	2
Short answer	10	4
Problem	3	15

Note that a total of 105 points are possible. Your grade will be computed out of 100 points (that is, if you earn 100/105 points, you will have a perfect score on the exam).

The exam will be planned to take around 90 minutes to complete; however, you will have the full 110 minute exam period in which to work.

You will be allowed to bring *one* 8.5" x 11" sheet of paper containing your notes. You may write on both sides of the paper.

Sample questions (more to come!)

Multiple choice:

1. What phrase applies *least* to the IP service model?
    a) Unordered delivery
    b) Best effort delivery
    c) Reliable delivery
    d) At-most-once delivery

2. The purpose of flow control is to
    a) Prevent the sender from sending too fast for the network to handle
    b) Prevent the sender from sending too fast for the receiver to handle
    c) Prevent the sender from having to retransmit data
    d) Prevent the sender from getting duplicate acknowledgments

Short answer:

1. Why might we have error detection both at the link layer and at the application layer?
2. Give two reasons why link state routing does NOT scale to the entire Internet.

Problem:

Problems from homework and quiz section give the best idea of what to expect from problems on the final. Here's a representative example (Peterson 6.31).

Suppose two TCP connections share a path through a router R. The router's queue size is 6 segments; each connection has a sliding window of 3 segments. No congestion control is used by these connections. A third TCP connection now is attempted, also through R. The third connection does not use congestion control either.

a) Describe a scenario in which, for at least a while, the third connection gets none of the available bandwidth, and the first two connections proceed with 50% each.
b) Does it help if the third connection uses slow start? Why or why not?
c) How does full congestion avoidance on the part of the first two connections help solve this?

CSE/EE 461 topics as of 2/25/04

Reliability: How do we building a network out of unreliable, distributed components?

- Coding: NRZ, Manchester, 4B/5B, XOR
- Framing: bit stuffing and alternatives
- Error detection & correction: Hamming distance, 2D Parity, checksums, CRCs, retransmission vs forward error correction
- Model of a link: bandwidth and delay
- Retransmission: ARQ, stop-and-wait, sliding window, flow control
- Retransmission timers: RTT estimation, Karn/Partridge algorithm, Jacobson/Karels algorithm
- Connections: the transport layer, ports, demultiplexing, three-way handshake, connection teardown, TCP state transitions
- Learning bridges & spanning tree algorithm; soft state & learning
- Routing: network as a graph
    - Distance vector; counting to infinity; split horizon & poison reverse
    - Link state; convergence, LSP flooding, Dijkstra's algorithm; cost metrics
- Virtual circuits vs datagrams; hard state vs soft state
- End-to-end argument; careful file transfer

Heterogeneity: How do we support a range of media and applications?

- Layering and protocol stacks; Internet and OSI models
- Internet design goals
- IP hourglass
- IPv4 header
- ICMP
- Maximum Transmission Unit (MTU) problem
- Sockets
- HTTP

Resource sharing: How do we share resources without centralized control?\

- Medium access control: goals, fixed allocation schemes vs statistical multiplexing
- Ethernet: carrier sense multiple access (CSMA) collision detection, exponential backoff
- Token ring: basic idea, comparison to Ethernet
- Wireless: hidden and exposed terminals, collision avoidance, RTS/CTS protocols
- Bandwidth allocation: congestion, fairness, design taxonomy
- Scheduling and buffer management
- TCP congestion control: TCP sawtooth, self-clocking, AIMD, slow start, fast retransmit, fast recovery
- Congestion avoidance: Random Early Detection (RED), Explicit Congestion Notification (ECN), basic idea of TCP Vegas
- Quality of service: application taxonomy, jitter, IntServ, token buckets, RSVP, fair queuing, DiffServ

Growth & Evolution: How do we cope as networks grow and evolve over time?

- Interdomain routing: why link state doesn't scale, hierarchy, structure of the Internet, path vector, policy
- Addressing: hierarchy, IP address classes, subnets, supernets (CIDR)

Fishnet

- Flooding, link state routing, forwarding, sliding window protocol