CS168 Fall 2014
Final Practice

Name: ______________________________

SID: ______________________________

Discussion Section (Day/Time): ______________________________

Instructions

In the actual final, questions will be roughly arranged in order of difficulty. This is not true for these practice questions, which are arranged arbitrarily.
Also, these practice questions aren't necessarily representative of the length, difficulty, or content of the final, but they do draw heavily from last year's midterm.

---

1. (- points) **Miscellaneous Questions**

   (a) (- points) **Five Basic Design Decisions** The Internet architecture was shaped by five basic design decisions. Please list the two in the following list that are NOT among these five decisions.

   - ◯ Layering
   - ◯ Longest prefix match
   - ◯ Best-effort service
   - ◯ The end-to-end principle and fate sharing
   - ◯ A single universal internetworking layer
   - ◯ Sliding window flow control
   - ◯ Packet switching

   (b) (- points) **ARP** A typical ARP exchange includes the following messages. Which of these messages are broadcast?

   - ◯ Initiating host sends: ARP request
   - ◯ Responding host sends: ARP response

   (c) (- points) **Netmask** Which of the following methods are ways a host can learn the netmask for the subnet?

   - ◯ Configuration
   - ◯ ICMP
   - ◯ ARP
   - ◯ DHCP
   - ◯ NAT

(d) (- points) **Headers** You are accessing a web site using your browser, from a host that is connected to an Ethernet within Soda Hall. A packet sniffer on the Soda Hall Ethernet captures a packet from your web session, which has TCP, IP, HTTP and Ethernet headers: starting from the outermost header (the header with bits at the very front of the packet), what is the order of the headers you need to traverse before reaching the payload?

(e) (- points) **CIDR** The number of addresses represented by the IP prefix 12.3.4.0/24 is: (Circle one)

- ◯ 64
- ◯ 256
- ◯ 16 million
- ◯ 6

(f) (- points) **Routing Protocols** Routing protocols such as RIP (distance-vector) and OSPF (link-state) are typically implemented at: (Circle one)

- ◯ The control processor of each router in a domain
- ◯ The control processor at only the border routers of a domain
- ◯ The line-cards of each router in a domain
- ◯ None of the above

(g) (- points) **Routers** The problem with input queued routers is: (Circle one)

- ◯ They require an interconnect fabric with capacity N times the output line rate (where N is the number of router ports)
- ◯ They suffer from head-of-line blocking

- ◯ They require extra memory for buffering packets
- ◯ They require complicated packet classification capabilities

(h) (- points) **IPv6** In IPv4 routers may be expected to fragment packets that exceed the link MTU (max. transmission unit). In IPv6, however, routers do not implement fragmentation. The rationale for this change is an example of which principle in action: (Circle one)

- ◯ Modularity through layering
- ◯ The end-to-end argument
- ◯ Robustness by eliminating state in routers

(i) (- points) **Packet Switching** Packet switching requires routers to maintain state for each TCP connection (Circle one)

- ◯ True
- ◯ False

(j) (- points) Four flows request bandwidth allocations of 1Mbps, 4Mbps, 8Mbps and 12Mbps. The capacity of the link is 12Mbps. Which of the following allocations are max-min fair?: (Circle one)

- ◯ 3Mbps, 3Mbps, 3Mbps, 3Mbps
- ◯ 2Mbps, 3Mbps, 3.5Mbps, 3.5Mbps
- ◯ 1Mbps, 3.66Mbps, 3.66Mbps, 3.66Mbps
- ◯ 1Mbps, 4Mbps, 7Mbps, 0Mbps

2. (- points) **Design**

   (a) (- points) You are the provider of a news channel that serves short 100Byte messages to a user population that is evenly spread across the globe. You plan to build a service in which a client opens a connection to one of your servers, requests one 100Byte message and terminates the connection after it has received the message. You get lots of (sometimes conflicting) advice on how to build a system that delivers messages both reliably and quickly. Assume the TCP MSS and network MTU are greater than 100Byte and that the size of the request is less than one MSS. Circle the advice that you agree with: (Circle all that apply)

   - ◯ It is preferable to invest in k servers spread across the globe each with a bandwidth serving capacity of C, than in a single server with a bandwidth serving capacity of kC
   - ◯ Sign up with multiple provider ISPs; for each user U, route messages to U via the provider ISP that advertises the shortest BGP path to U. This is guaranteed to ensure packets reach U with the shortest delay.
   - ◯ Your users will enjoy faster downloads if you make the following adjustment to your TCP implementation: trigger fast retransmits after one (instead of the standard three) duplicate ACK
   - ◯ The default value for a TCP timeout is 500ms. Setting the default to 100ms can only improve a user's download speed. (Ignore any additional overhead introduced at the server due to the potentially larger number of timeouts)
   - ◯ The packet loss rate of the network is less than 1%. So, don't bother with TCP and instead just build your application on top of UDP. Clients send one UDP packet to request a message and your server responds by transmitting the requested message thrice (using three separate UDP packets). If the client does not receive the message, it will simply resend its request. On average, this approach will result in faster downloads for users and will transmit no more packets than TCP would.

   (b) (- points) You are a router vendor charged with developing the next-generation high-speed router. Consider each of the proposed architectural changes listed below and determine whether it makes your task **easier, harder, or has no impact** on your task.

   (1) Eliminate options from the IP header

   (2) Switch to a rate-based congestion control protocol such as RCP

   (3) Endhosts have infinite buffer memory at the transport layer

   (4) Replace UDP for TCP as the de-facto transport protocol

   (5) Packets carry a "source route". A source route is a fixed-length ordered list of all the routers along the path between a source and destination. [Hint: consider how a router performs a forwarding lookup given a source route.]

3. (- points) **HTTP and Caching**

   (a) (- points) Consider a case where a client A is retrieving files F and G from web site B. F and G are both 125KB (i.e., one megabit). The RTT between A and B is 10msec (note, these are round-trip-times, not one-way latencies), and the bandwidth between the sites is 10Mbps. Assume all TCP SYN/ACK packets and HTTP request packets are negligible in size. How long does it take A to retrieve both files under each of the following circumstances?

      1. Sequential, nonpersistent TCP connections
      2. Concurrent, nonpersistent TCP connections
      3. Sequential, 1 persistent TCP connection
      4. Pipelined, 1 persistent TCP connection

   (b) (- points) Consider the same situation as in (i), but assume that rather than a dedicated link there is a large shared link with many flows traversing it, and each TCP connection gets 10Mbps (adding additional flows does not significantly change the bandwidth per TCP connection, because there are thousands of flows on the link). Now, how long does it take A to retrieve both files under each of the following circumstances:

      1. Sequential, nonpersistent TCP connections
      2. Concurrent, nonpersistent TCP connections
      3. Sequential, 1 persistent TCP connection
      4. Pipelined, 1 persistent TCP connection

   (c) (- points) Consider the same situation as in i), except that A is only downloading file F and there is now a cache C between A and B. All requests from A to B go through cache C, and assume the bandwidth along the path from A to C is 1gbps and the RTT between A and C is negligible, while the bandwidth along the path from C to B is 10mbps with an RTT of 10msec. Note, these are round-trip-times, not one-way latencies. As above, assume that the file is 125KB (i.e., one megabit) and that all TCP SYN/ACK packets and HTTP request packets are negligible in size. Assume the cache operates as follows: (where the origin server refers to the site named in the URL)

      - If the object is not in the cache, the request is forwarded to the origin server
      - If the object is in the cache, and the cache entry has not timed out (i.e., the cache TTL has not expired), the object is returned to the client

- If the object is in the cache, but the cache entry has timed out, the cache issues a conditional-GET to the origin server, asking if the object has changed since this object was cached: if the origin server responds that it hasn't, the cache returns the cached object, otherwise the origin server responds with the updated object which the cache forwards to the client.

How long does it take for A to receive the file under the following circumstances:

1. File is not in the cache
2. File is in the cache, TTL has not expired
3. File is in the cache, TTL has expired, file is unchanged
4. File is in the cache, TTL has expired, file has changed

4. (- points) **Wireless** We consider three collision resolution schemes:

   - Scheme X (pure carrier sense): Never send when you hear someone else transmitting, but otherwise can send whenever you want.
   - Scheme Y (classic MACA): No carrier sense. Nodes wishing to communicate use an RTS-CTS-Data-ACK exchange. Nodes overhearing an RTS wait to allow the CTS to be sent. If no CTS is heard, the node can transmit. If a CTS is heard (even if no earlier RTS is heard), the node is quiet for the entire duration of the data transmission.
   - Scheme Z (hybrid approach closer to 802.11): Carrier sense. Nodes overhearing either an RTS or a CTS are quiet for the entire duration of the transmission (data and ACK).

   We have four wireless nodes A, B, C, D, where A can only hear B (but not C or D), B can only hear A and C (but not D), C can only hear B and D (but not A) D can only hear C (but not A or B). A and B are in the midst of a communication, and C has been listening to their exchange so far (and so has heard whatever RTS's or CTS's B has sent so far). While A and B are in the "sending data" part of their exchange, C decides that it wants to communicate with D. Consider two cases:

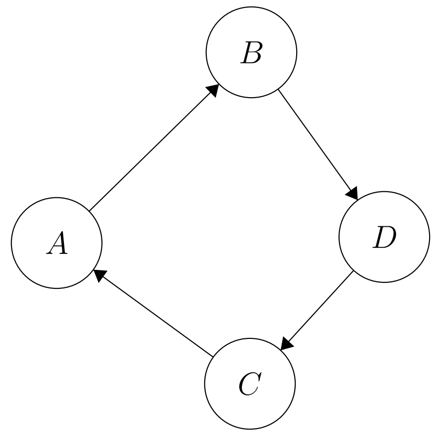
   (a) (- points) A sending to B

      - If scheme X is used, would C be allowed to send a message to D?
      - If scheme Y is used, would C be allowed to send a message to D?
      - If scheme Z is used, would C be allowed to send a message to D?

   (b) (- points) B sending to A

      - If scheme X is used, would C be allowed to send a message to D?
      - If scheme Y is used, would C be allowed to send a message to D?
      - If scheme Z is used, would C be allowed to send a message to D?

5. (- points) **Load Sensitive Routing**



Consider the above network (pretend the arrows are lines), where A sends traffic to D. Assume the routing protocol computes the shortest path, where the cost of a link is based on its traffic load (i.e., the amount of traffic on the link). The traffic from A to D is the only traffic in the network. Which route is traffic from A to D going to use? Is the route stable? Explain why or why not.

[While the initial starting point does not affect your answer, you may assume that the initial link costs (i.e., before traffic flows) equal one and that A breaks ties between equal cost paths by routing through the neighbor with the lowest ID (alphabetical order). You may further assume that the interval between routing updates (and hence changes in routes) is much longer than the path RTTs.]

6. (- points) **BGP** Say we have ASes A,B,C,D,E. A is a provider to B and D. B and D are peers. C is a customer of B and E is a customer of C and D.

   (a) What path, if any, would B advertise for destination E?

   (b) What path will A use to reach E? Is this necessarily the shortest path in terms of router hops? Why not?

   (c) If link D-E fails, what path will D use to reach E?

   (d) If link D-E fails, what path will A use to reach E?

   (e) If both A-D and B-D fail, can C use the path C-E-D to reach D? Why or why not?