CSE 473S – Introduction to Computer Networks – Fall 2006

Exam 1

Problems and Solutions

Problem 1 (**15 points**). Imagine that you start a browser, request *www.amazinglygreatnews.com*, and the returned page greets you by your correct name. While HTTP is stateless, briefly explain how the web server learned your name.

Solution: *The web server recognizers the user due to a cookie, a number identifying the user. The cookie is established during a previous session and stored at both server and browser. When the browser sends the request, the browser includes the cookie. Upon receiving the request, the server uses the cookie to find the user's name in the local database and puts the name into the response.*

Problem 2 (**15 points**). What happens after you issue *telnet cse.wustl.edu 21* command? What do you expect to see in reply?

Solution: *The command establishes a TCP connection to an FTP server. The server is expected to reply with a greeting that includes a three-digit code (220) and an optional text message following the number.*

Problem 3 (**20 points**). In the considered packet-switched store-and-forward network, a client requests an object by sending a packet of size $2L$, and a server supplies the requested object by sending a packet of size L. At time 0, client A requests two objects from server C via router B. Link AB has capacity $2R$ and propagation delay d. Link BC has capacity R and propagation delay $2d$. Nodal processing delays are negligible. There is no other traffic in the network. When does the client receive the first object? When does the client receive the second object?

Show your derivations. Use mathematical expressions, not their verbal descriptions. Simplify your final analytical expressions. Explain (briefly) your reasonings.

Solution: *Transmission delays for links AB, BC, CB, and BA are $t_{AB} = \frac{L}{R}$, $t_{BC} = \frac{2L}{R}$, $t_{CB} = \frac{L}{R}$, and $t_{BA} = \frac{L}{2R}$ respectively.*

The server receives the first request at time $t_1^C = t_{AB} + d + t_{BC} + 2d = \frac{3L}{R} + 3d$. The client receives the first object at time $t_1^A = t_1^C + t_{CB} + 2d + t_{BA} + d = \frac{9L}{2R} + 6d$.

Since link BC has a smaller capacity than link AB, the router starts forwarding the second request right after the router finishes forwarding the first request. Hence, the server receives the second request at time $t_2^C = t_1^C + t_{BC} = \frac{5L}{R} + 3d$. By time t_2^C, the server already finished transmitting the first object. Thus, the server starts transmitting the second object at time t_2^C, and the client receives the second object at time $t_2^A = t_2^C + t_{CB} + 2d + t_{BA} + d = \frac{13L}{2R} + 6d$.

Problem 4 (**15 points**). Port *2* of router *NSP1* in an ONL configuration has the following forwarding table:

prefix/mask	next hop
192.168.1.64/30	1
192.168.1.72/29	2
192.168.1.75/30	3
192.168.1.77/24	4
192.168.1.77/27	5
192.168.1.77/28	6
192.168.1.78/31	7

Port *2* receives a datagram addressed to *192.168.1.77*. Where does the port forward the datagram? Briefly describe the algorithm used by the port. Also, briefly explain how you applied the algorithm to find your answer to the above question.

Solution: *The router forwards the datagram to the external link leaving from port 2 because 192.168.1.72/29 is the longest prefix matching 192.168.1.77.* The last bytes of the matching prefixes are 01001*** (.72/29), ******* (.77/24), 010***** (.77/27), and 0100**** (.77/28).

Problem 5 (**15 points**). Imagine that you are a provider of a link from node B to node C. The average queueing delay at the link is always equal to $\frac{A}{R-A} \cdot 10$ ms, where A is the traffic load on the link, and R is the link capacity. The current values of the link queueing delay and capacity are respectively $Q = 40$ ms and $R = 100$ Mbps. To what value you should upgrade the link capacity in order to reduce the queuing delay to $q = 10$ ms?

An alternative solution is to install a cache at node C. How high should the hit ratio of the cache be to achieve the desired goal (of reducing the the queuing delay to 10 ms)?

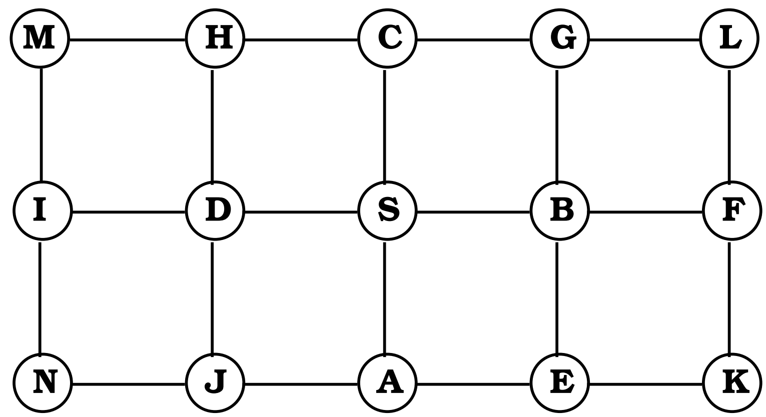
Show your derivations. Use mathematical expressions, not their verbal descriptions. Do not approximate. Explain (briefly) your reasonings.

Solution: *Currently* $Q = \frac{A}{R-A} \cdot 10ms$, *meaning that the current traffic load is* $A = \frac{Q}{Q+10\,ms} \cdot R = \frac{40\,ms}{40\,ms+10\,ms} \cdot 100\,Mbps = 80\,Mbps$. *The new link capacity* R_n *should be such that* $q = \frac{A}{R_n - A} \cdot 10$ *ms. Then,* $R_n = \frac{q+10\,ms}{q} \cdot A = \frac{10\,ms+10\,ms}{10\,ms} \cdot 80\,Mbps = 160\,Mbps$.

The alternative solution should reduce the link load to A_n *by serving an* h *fraction of the traffic from the cache, i.e.,* $q = \frac{A_n}{R-A_n} \cdot 10ms$ *and* $A_n = (1-h)A$. *Hence,* $A_n = \frac{q}{q+10\,ms} \cdot R = \frac{10\,ms}{10\,ms+10\,ms} \cdot 100\,Mbps = 50\,Mbps$, *and the needed hit ratio is* $h = 1 - \frac{A_n}{A} = 1 - \frac{50\,Mbps}{80\,Mbps} = \frac{3}{8}$ *or 37.5%.*

The direct expressions are $R_n = \frac{Q}{q} \cdot \frac{q+10\,ms}{Q+10\,ms} \cdot R = 160\,Mbps$ and $h = 1 - \frac{q}{Q} \cdot \frac{Q+10\,ms}{q+10\,ms} = \frac{3}{8} = 37.5\%$.

Problem 6 (**20 points**). Consider the following network where a browser and its local DNS server are located in node M, a web server is at node E, and the DNS hierarchy consists of root server L, top-level domain (TLD) server B, and authoritative server K:



From time to time, the browser retrieves short HTML files (each fits within one packet; every HTTP request is even smaller) from the web server. The involved application-level protocols employ their default transport service. HTTP uses nonpersistent connections. All stages of DNS querying are recursive. Caches at the intermediate DNS servers have the following *hit* ratios: $h_1 = 0.5$ for the local server, $h_2 = 0$ for the root server, and $h_3 = 1$ for the TLD server. Each exchange of messages follows a path with the smallest delay between the communicating nodes. Each link has propagation speed $s = 300,000$ km/sec and length $l = 900$ km. Nodal processing, queueing, and transmission delays are negligible. What is the average response time, i.e., the average time it takes for the browser to receive an HTML file?

Show your derivations. Explain your reasonings.

Solution:

Each link has propagation delay $\frac{l}{s} = 3ms$.

DNS query:
Case 1 (h_1 *or 50% of the time): instantaneous hit; 0 hops; query time* $t_h = 0$.
Case 2 ($1 - h_1$ *or the other 50% of the time): query path* $M - L - B - L - M$*, 12 hops, query time* $t_m = 36ms$.
The average query time is $t_q = h_1 t_h + (1 - h_1) t_m = 18ms$ *(or 6 hops).*

Establishing a TCP connection from the browser to the server: 10 hops; time $t_c = 30ms$.

Retrieving the file itself: 10 hops; time $t_f = 30ms$.

The average response time is $t_r = t_q + t_c + t_f = 78ms$ *(or 26 hops).*