CS 421: Computer Networks **FALL 2006**

MIDTERM I
November 8, 2006
120 minutes

Name: ______________________________

Student No: ___________________________

Show all your work very clearly. Partial credits will only be given if you carefully state your answer with a reasonable justification.

Q1	
Q2	
Q3	
TOT	

1)

a) (6 pts) HTTP protocol is stateless. Describe what exactly that means. Is FTP also stateless? Why or why not?

b) (6 pts) If HTML is replaced by another markup language, e.g., XML, will HTTP require any modifications? Justify your answer.

c) (6 pts) What type of applications does not use TCP? Why? Give all relevant reasons.

d) (6 pts) Why does a TCP client stay in the TIME_WAIT state for a fixed amount of time before moving to the CLOSED state? How is this fixed time determined?

e) (6 pts) How is the EstimatedRTT updated in TCP when a segment is retransmitted? Explain the reason for this rule.

f) Consider two packets sent by two different browser applications to the same Web server. Complete the sentences in each of the following questions with one of the following phrases

- are always same
- can be different or same
- are always different

and then briefly justify your answer.

i) (3 pts) Consider the destination IP addresses of these two packets. They ...

Answer:

Justification:

ii) (3 pts) Consider the destination port numbers of these two packets. They ...

Answer:

Justification:

iii) (3 pts) Consider the pair composed of (source IP address, source port number) of these two packets. They ...

Answer:

Justification:

2)

a) Suppose that you are required to use the Stop-and-Wait protocol for reliable communication over a 100 Mbps (100 x 10^6 bps) channel with an end-to-end distance of 6000 km. Each packet sent by the sender is 1000 bytes long. Assume that the speed of propagation is 3 x 10^5 km/s and ignore all other sources of delay other than the propagation delay. ACK packets have negligible transmission times.

   i) (5 pts) What is the utilization of the channel, U, which is defined as the fraction of time the sender is busy sending bits into the channel?

   ii) (5 pts) Assume now that a computer hacker changed the receiver side of the protocol such that the receiver is sending periodic ACKs totally independent of the packets it is receiving. Suppose that the receiver is sending an alternating sequence of ACK 0 and ACK 1, where an ACK (ACK 0 or ACK 1) is sent every 1 msec. What is the average delay from the completion of the transmission of the packet until the ACK with the correct sequence number is received, i.e., the sender can now send a new packet. What is utilization of the channel, U? Compare with your answer in i).

   iii) (5 pts) Is there any problem with this modified Stop-and-Wait protocol?

b) (10 pts) Consider a 1 Mbits/sec channel with a 20 msec one-way propagation delay, i.e., 40 msec roundtrip propagation delay. We want to transfer a file of size 13500 Bytes. Each packet has a total size of 1625 Bytes including the 125 Bytes header, i.e., each packet contains 1500 Bytes of data. When there is data to be transmitted, each packet contains the maximum number of bytes. Assume that ACK packets are of 125 Bytes long and there is a processing delay of 1 msec after a packet is fully received at the receiver until the transmission of the corresponding ACK is started. **Selective Repeat** protocol is used with a window size of N = 4 packets. Assume that every 6^{th} packet crossing the forward channel is lost while ACKs are not lost or corrupted. Assume that the processing delay at the sender after an ACK is received is negligible. How much time is required to complete the transfer of the whole file and receive the final ACK at the sender? Assume that the timeout for each packet is set to 3msec larger than the **total** round-trip delay (including transmission, propagation and processing delays).

c) (8 pts) Consider a transmission link running at 1 Gbps (1 x 10^9 bps). Assume that the speed of propagation is 3 x 10^5 km/s. What is the maximum distance between a TCP source and destination so that the window can be fully utilized. Note that the receive window field in the TCP header is 2 bytes long.

3)

a) We would like to transfer a 4 Mbytes file using TCP. Assume that the sender is in the slow start phase at the beginning of the file transfer and the slow start threshold is set to 10 Mbytes. Assume further that TCP has been modified to allow very large windows so that the TCP receive window is always 2 Mbytes during the entire connection. The distance between the sender and the receiver is 3000 km. Assume that the speed of propagation is 3×10^5 km/s. Each TCP segment has a size of 1 Kbytes. The transmission speed of the connection is 100 Mbps (100×10^6 bps). For simplicity, ignore the segment transmission time, processing and store and forward delays.

i) (5 pts) Suppose no packets are lost. How long does it take to transfer the entire 4 Mbytes file?

ii) (5 pts) What is the average channel utilization, U, during the transfer of the file?

b) Suppose a TCP connection experiences round-trip times (RTT) of 20 msec for 60% of its packets, 40 msec for 30% of its packets and 100 msec for 10% of its packets. Suppose no packets are actually lost. Assume that the estimated RTT (according to the exponential weighted moving average) is equal to the true (ensemble) average of RTT, i.e., EstimatedRTT = average value of RTT = E[SampleRTT].

i) (5 pts) Assume that the timeout is set to **2 (two)** times the estimated RTT (as in the original version of TCP), i.e., TimeOut = 2 x EstimatedRTT. What fraction of the packets will be assumed lost by the TCP sender?

ii) (5 pts) Currently used versions of TCP estimate both the mean and the mean deviation as we discussed in the class (the mean deviation is the average absolute distance of RTT samples from the estimated RTT), and sets the timeout to the estimated mean (estimatedRTT) plus **4 (four)** times the estimated deviation (devRTT), i.e., TimeOut = EstimatedRTT + 4 x devRTT. Assume that the TCP connection uses this new method, and the estimated RTT and deviation are equal to their true (ensemble) values, i.e., EstimatedRTT = E[SampleRTT] and devRTT = E[|SampleRTT-EstimatedRTT|]. What fraction of the packets will be assumed lost by the TCP sender?

c) (8 pts) Assume a congestion feedback model for a system composed of two flows sharing a bottleneck link with bandwidth R bits/sec where both connections have the same RTT. Each flow gets binary synchronous feedback in discrete time steps of one RTT. If the aggregate consumption of the two flows is above the bottleneck bandwidth, both senders receive a congestion notification signal (**CN**), otherwise they receive no CN. The flows use a simple congestion control scheme: When no CN is received in a time step, each sender increases its window by one segment. On the other hand, for each congested time step, i.e., when both senders receive CN, each sender decreases its window by one segment. So, we can call this algorithm as *Additive Increase-Additive Decrease* (**AIAD**). **Prove or disprove** that AIAD achieves a fair allocation of bandwidth between the flows, i.e., each flow getting R/2, by using graphical arguments similar to the one we made in class in showing that TCP's AIMD algorithm is fair.