Take-Home Quiz I
Due Tuesday March 3 11:55pm

Csci4211: Introduction to Computer Networks
Spring 2020
Prof. Zhi-Li Zhang

---

Instructions:

1. This is an **open-book** and **open-note** quiz.
2. There are **four** big questions in total, each of which has several sub-questions. You have more than **two days** to answer the questions.
3. Partial credit is possible for an answer; please include intermediate steps as appropriate. Please try to be as concise and make your exam as neat as possible. We *must* be able to read your handwriting in order to be able to grade your exam.
4. **Please work on the quiz *individually, by yourself only!* No discussion among the students in the class, or with others, is allowed. If you find your answers from the Internet or any other sources, please cite your sources. Any violation of the University's *Student Conduct Code* will be reported to both the department and the University, and you may be suspended or expelled! Please note that if you let another student copy your answers, you are also in violation of the University's *Student Conduct Code*.**
5. Please make sure to write down your name and student id. on your answer sheets. Please combine your answer sheets in a *single* file (preferably in pdf or in MS word doc if you don't have a tool to convert your file into pdf), and submit it *electronically via the class Canvas site.*
6. Good luck. Enjoy!

---

## 1. True or False with a Brief Explanation: (15 points total. approx. 15 minutes)

*For each* **true or false** *question, please record your answer (*`T` *or* `F`*)* with one or two sentences briefly justifying your answer!

**a.** (3 points) **True or False:** Packet switching is faster than circuit switching because it does not require connection set-up.

**b.** (3 points) **True or False:** Circuit-switched networks do not incur any propagation delay.

**c.** (3 points) **True or False:** HTTP/1.0 is *statelesss* while HTTP/1.1 is *stateful* because the latter requires a *persistent* TCP connection.

**d.** (3 points) **True or False:** The SMTP protocol used in email applications requires that the sender and recipient email addresses included in the email message body – i.e., in the email text after the "`DATA`" command and before the " `.`" command (e.g., those in the "`From:`" and "`To:`" header fields of the MIME message format) – must be the same as those used in the SMTP commands "`MAIL FROM:`" and "`RCPT TO:`".

**e.** (3 points) **True or False:** The *checksum* field carried in the TCP/UDP header allows the recipient of the packet to check for *any kind of bit errors* that occur during the data transmission.

## 2. DNS and UDP vs. TCP (36 points total. Approx. 25 minutes)

**a.** (4 points) There are (at least) 13 (logical) root DNS servers distributed across the globe. What are the possible *disadvantages* of deploying a single (physical or logical) root DNS server placed at one location in the world, say, Chicago in US.

**b.** (4 points) When your laptop sends a DNS query to the local DNS server, it sets the *recursion desired* flag in the DNS query message and asks the local DNS server to perform recursive queries. What is the advantage of using the recursive DNS query at the local DNS server? What could be a possible disadvantage?

**c.** (4 points) Suppose the *propagation delay* from your laptop to the local DNS server (and vice versa) is 1 ms. What is the *minimal* time it takes for your laptop to receive a DNS response for the DNS query it has just issued to the local DNS server? [For this and the following questions, you can assume that the transmission or other delays are negligible (i.e., zero!).] Briefly justify your answer.

**d** (10 points) For the same question in **c.**, suppose your laptop is querying for `www.xyz.com`, and your local DNS server does not have an answer cached for this query. The minimal *round-trip-delay* from your local DNS server to the closest root DNS server is 20 ms, the minimal *round-trip-delay* from your local DNS server to the closest top-level domain (TLD) DNS server `.com` is 10ms, and the minimal *round-trip-delay* from your local DNS server to the authoritative DNS server `dns.xyz.com` is 40ms. How long will it take for your laptop to receive a DNS response from your local DNS server? Please justify your answer (draw a figure if needed to illustrate your answer).

**e.** (14 points) Consider the question in **c.** again, but suppose that instead of UDP, TCP is used to deliver DNS queries and responses between your laptop and the local DNS server. What is the *minimal* time it takes for your laptop to receive the DNS response for the DNS query it has just issued to the local DNS server, **i.)** if the answer is cached at your local DNS server; and **ii.)** if the answer is *not* cached at your local DNS server? Please justify your answers (draw figures if needed to illustrate your answer).

## 3. TCP Connection Management (24 points total. Approx. 20 minutes)

The following figure shows the control messages sent among the client and the server under normal operations using the *three-way handshake* protocol. (Note: in `SYNACK(y,x)` and `ACK(x,y)`, the first number is the sequence number of the message, the second number is the acknowledgment number, i.e., the sequence number of the message being acknowledged.)

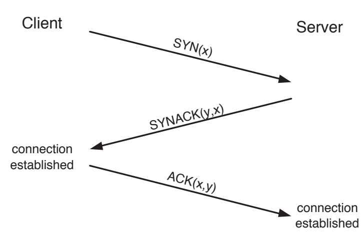


Figure 1: TCP 3-way handshake.

**a.** (6 points) Consider the following scenario (see Figure 4) where the `SYN(x)` message sent by the *client* is lost during the transmission. What will happen at either the client or the server side?

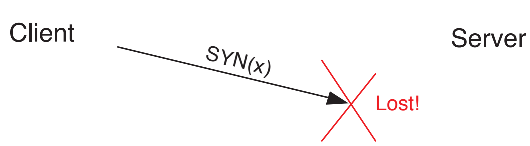


Figure 2: Figure for Question 3a.

**b.** (6 points) Now consider another scenario as illustrated in Figure 3 where the `SYNACK(y,x)` message sent by the *server* is lost during the transmission. What will happen at either the client or the server side?

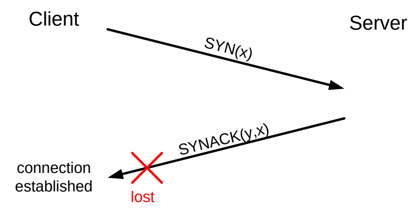


Figure 3: Figure for Question 3b

**c.** (6 points) Suppose that by now the connection in **a.** or **b.** (where the client used the initial sequence no. `x`, and the server used the initial sequence no. `y`) has been closed. An old, duplicate message `ACK(x,y)` now pops up at the **server** side (see Figure **??**). First, can this scenario happen at all? Second, in response to this `ACK(x,y)` message, what will the *server* do? Briefly explain your answers to both questions.

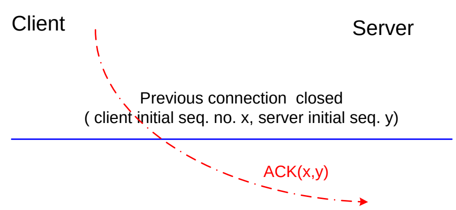


Figure 4: Figure for Question 3c.

**d.** (6 points) Briefly explain why TCP employs the *timed wait* mechanism when tearing down a (duplex) TCP connection. [See, e.g., Slides 27-28 of the lecture notes "Transport Layer Part 1", where the client side sends the first `FIN` message to indicate to the server that it has no more data to transmit; and when it receives the `FIN` message from the server, the client must enter into the `TIME_WAIT` state before fully closing the duplex TCP connection.]

## 4. Reliable Data Transfer and Protocol Efficiency (25 points total. Approx. 25 minutes)

**a.** (12 points) Consider the scenario in Figure 5 where the sender and the receiver are *directly connected by a duplex physical link*, and they employ the Go-Back-N protocol for reliable data transfer across the link. Assume that the sequence numbers are represented with 3 bits, and a window size of $W = 7$ is used in the Go-Back-N protocol. Figure 5 gives a snapshot of the data transfer, where the sender's and receiver's windows at the moment (*right before the sender starts transferring in this round*) are shown. The sender *then* sends the 7 packets as allowed by the sender's window shown in Figure 5. The receiver receives 5 of them, where the packets with sequence no.4 and sequence no.5 are lost during the transmission.

(i) (8 points) Describe the action at the receiver. In particular, specify the sequence no. which is cumulatively acknowledged by the receiver and the receiver's window after the reception of these packets.

(ii) (4 ponts) Suppose the acknowledgment sent by the receiver is lost during the transmission, what will eventually happen at the sender?

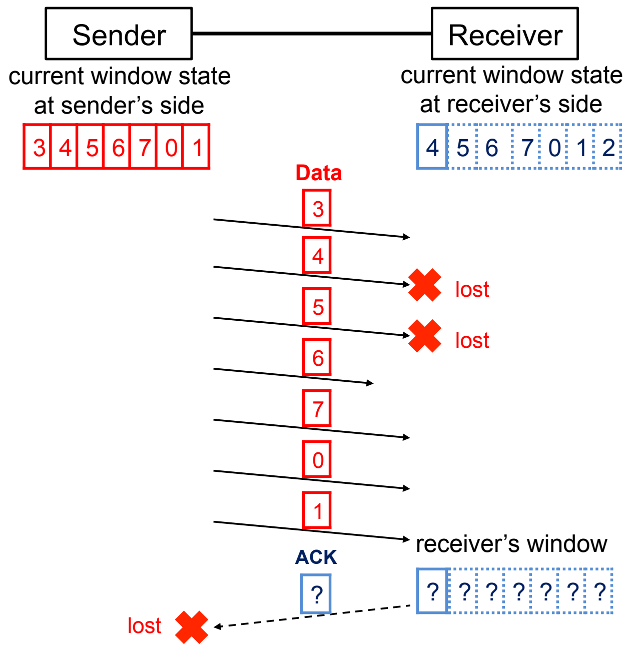


Figure 5: Figure for Question 4a.

**b.** (13 points) Consider the scenario in Problem **4.a**, where the sender and the receiver are *directly* connected by a duplex physical link, and they employ the Go-Back-N protocol for reliable data transfer across the link, with a window size $W = 7$. Suppose that the data rate of the link is 10Mbps, and the propagation delay is 5.6 milliseconds (ms). The data packets transmitted across the link is of fixed size: 1 KB. (For simplicity of calculation, in the following, please assume that 1 Mpbs $= 10^6$ bits per second, and 1 KB$= 10^3$ bytes. Also note that 1 byte (B) $= 8$ bits, and 1 millisecond (ms) $= 10^{-3}$ second. You can assume that the time to transmit the acknowledgment packets are negligible, namely, 0.) Please answer the following questions.

i) (8 points) What is the maximum utilization of the link?

ii) (3 points) If we want to *fully utilize* the link (i.e., with 100% utilization), what should the window size be? And iii) (2 points) what is the minimum number of bits we need to represent the sequence number?