Name_____ Student ID_____ Department/Year_____

2nd Examination

Introduction to Computer Networks (Online)
Class#: EE 4020, Class-ID: 901E31110
Spring 2025

10:20-12:10 Wednesday
April 30, 2025

Cautions

1. There are in total 100 points to earn. You have 90 minutes to answer the questions. Skim through all questions and start from the questions you are more confident with.
2. Use only English to answer the questions. Misspelling and grammar errors will be tolerated, but you want to make sure with these errors your answers will still make sense.

1. (ch26, 5pt) Recall how DASH server and client work. Tell which of the functions below are the jobs of the server and which of the client. **Grading policy: -1pt per wrong pick till 0pt.**

   (a) Measuring server-to-client available bandwidth
   (b) Maintaining the manifest file containing the URLs to different video chunks and rates
   (c) Encoding video chunks into different resolutions
   (d) Dividing videos into chunks
   (e) Requesting the manifest file containing the URLs to different video chunks and rates
   (f) Choosing the max rate of the video chuck at different points in time
   (g) Deciding when to pause requesting video chunks
   (h) Deciding when to resume requesting video chunks

   (1) Which are the jobs of a DASH server? (2pt)
   (2) Which are the jobs of a DASH client? (3pt)

2. (ch26, 6pt) Recall these two terms introduced in video streaming – buffer starvation and buffer overflow.
   (1) What is buffer starvation? (1pt)
   (2) How to prevent buffer starvation? (2pt)
   (3) What is buffer overflow? (1pt)
   (4) How to prevent buffer overflow? (2pt)

3. (ch26, 4pt) Due to the nature of video content and the high user demand, a video service today relies on a content distribution network (CDN) to deliver video steams within a reasonable amount of time. There are two types of CDN design. One is called (a)“enter deep” and the other (b)“bring home”.
   (1) Which type of the CDN designs tends to deploy servers closer to the viewers? (1pt)
   (2) Which type of the CDN designs tends to deploy more servers? (1pt)
   (3) Which design would you use if you are starting up a video service with friends and why? (1pt)
   (4) Which design would you use if you are working for a large video service (such as Youtube) and looking to expand the viewership any why? (1pt)

4. (ch3, 5pt) Recall the following functions of the transport layer protocols. **Grading policy: -**

1pt per wrong pick till 0pt.

(a) bit error detection
(b) packet loss detection
(c) bit error recovery
(d) packet loss recovery
(e) multiplexing
(f) demultiplexing
(g) byte counting
(h) security

(1) Which are implemented in UDP? (2pt)
(2) Which are implemented in TCP? (3pt)

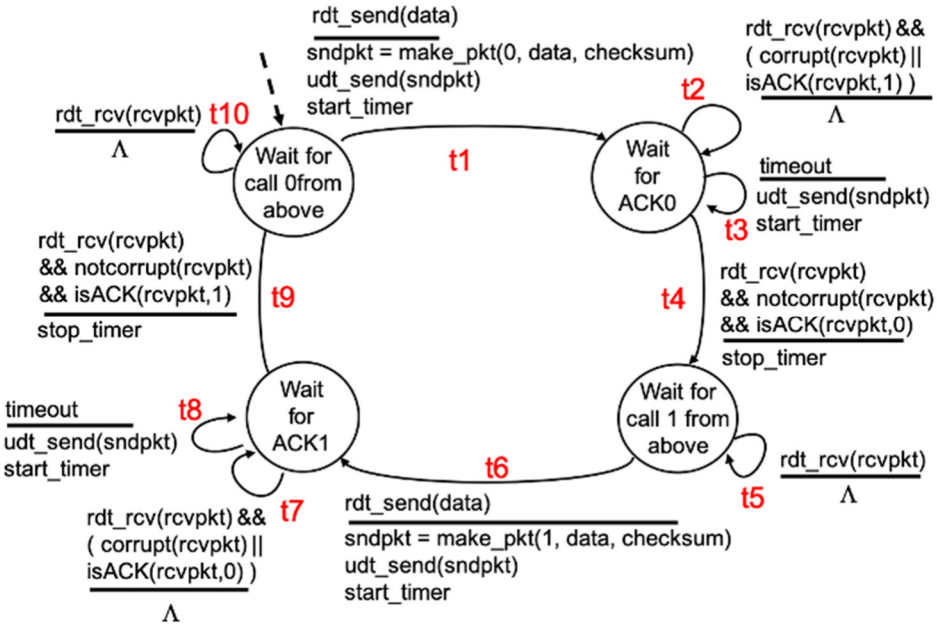
5. (ch33, 5pt) Recall how Internet checksum works. Calculate the checksum for the following 32-bit data and see if you can come up with bit error patterns that the Internet checksum fails to detect.

1010 1010 1010 1010 0101 0101 0101 0101

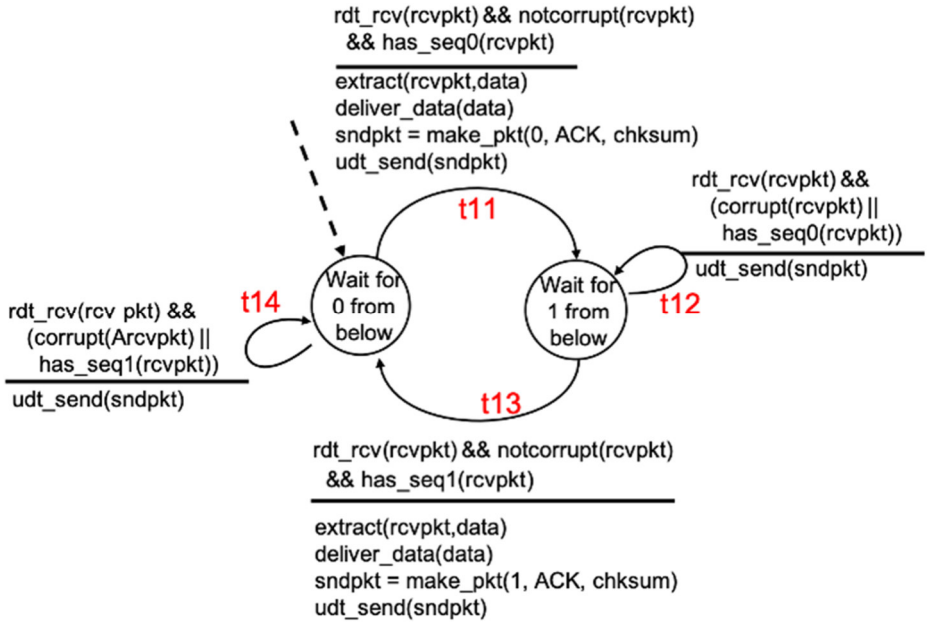
(1) Tell the checksum of the above data sequence. (1pt)
(2) Find a 32-bit data containing 2 bit errors that is not detectable by the Internet checksum. (1pt)
(3) Find a 32-bit data containing 4 bit errors that is not detectable by the Internet checksum. (2pt)
(4) Do you think Internet checksum will catch all bit error patterns and why? (1pt)

6. (ch34, 10pt) Below is the FSM of rdt 3.0. Tell the sequence of transitions for the following scenarios.

rdt 3.0 sender



rdt 3.0 receiver



(1) Both the sender and receiver start from the initial state. The sender sends one data packet. The packet arrives at the receiver but “ACK 0” returning to the sender has a bit error. There is no more bit errors or packet losses afterwards. (2pt)

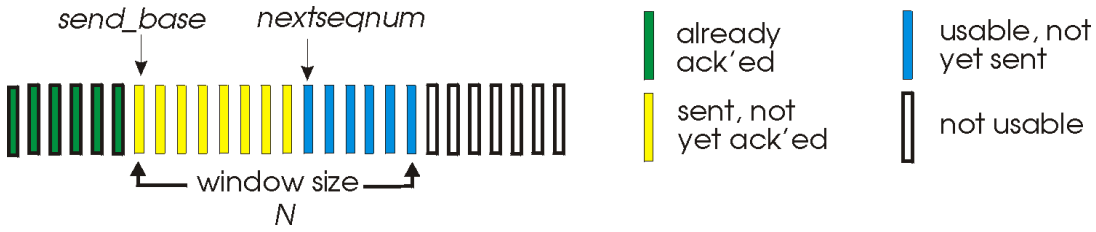
(2) Continue from (1). The sender sends one more packet. The packet arrives at the receiver but “ACK 0” returning to the sender has a bit error. There are no bit errors or packet losses afterwards. (2pt)

(3) Continue from (2). The sender sends one more packet. The “ACK 0” packet coming back is lost. There are no more bit errors or packet losses afterwards. (2pt)

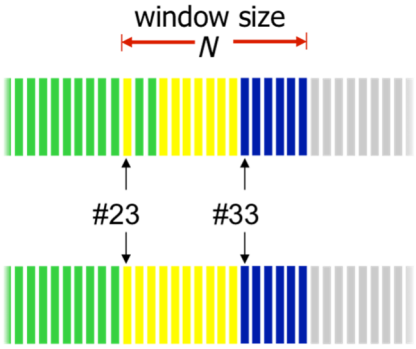
(4) Continue from (3). The sender sends one more packet. The data packet is lost. There are no more bit errors or packet losses afterwards. (2pt)

(5) In rdt 3.0 sender FSM, t2 does not do anything when the ACK packet received is corrupted or a duplicate. Do you like this? (0pt) Why or why not? (2pt)

7. (ch34, 4pt) Recall the color scheme of the packets at the sender side of GBN and SR.



Below are sending windows of two pipelined rdt connections. We know that one of them is a GBN connection and the other an SR connection. Tell the value of send_base when an ACK packet arrives back at the sender.



(1) ACK #23 arriving back at the GBN sender

(2) ACK #23 arriving back at the SR sender

(3) Continue from (1). ACK #26 arriving back at the GBN sender

(4) Continue from (2). ACK #26 arriving back at the SR sender

8. (ch35, 5pt) Compare and contrast Go-Back-N (GBN), Selective Repeat (SR), and Transmission Control Protocol (TCP). **Grading policy: -1pt per wrong pick till 0pt.**

(1) Which method(s) uses a fixed window size? (1pt)

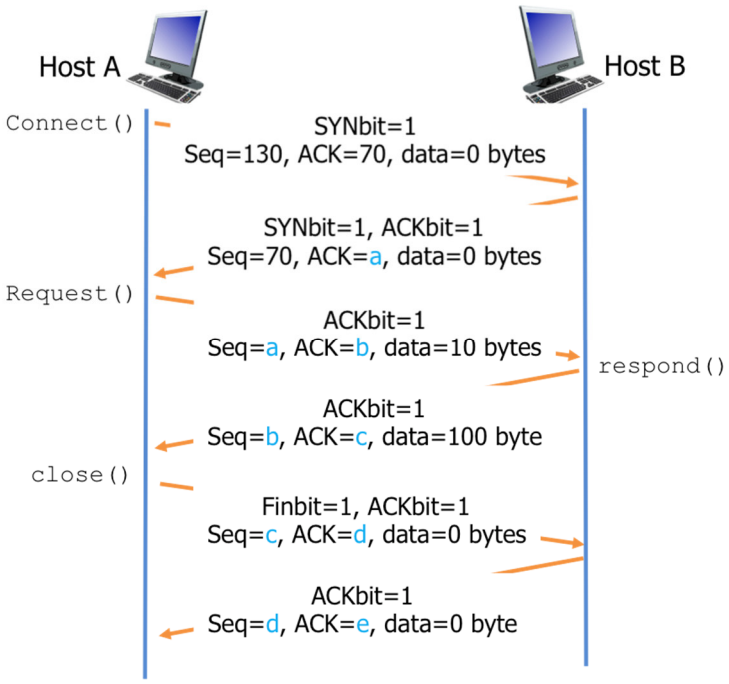
(2) Which method(s) uses the least amount of network bandwidth to retransmit for losses? (1pt)

(3) Which method(s) uses individual acknowledgements? (1pt)

(4) Which method(s) uses at most one timer at the sender end? (1pt)

(5) Which method(s) buffers out-of-order packets at the receiver end? (1pt)

9. (ch35, 5pt) Depicted below is a sequence of packet exchange in a TCP connection between Host A and B. Host A sends the SYN packet to establish connections and then generate request messages. Later, A sends a FIN packet to initiate closing of the connection. Tell the value of a, b, c, d and e.



10. (ch36, 5pt) There are two general approaches to congestion control – end-to-end and network-assisted.
    (1) Describe how the end-to-end approach works. (1pt)
    (2) Describe how the network-assisted approach works. (1pt)
    (3) Tell one advantage of taking the end-to-end approach. (1pt)
    (4) Tell one advantage of taking the network-assisted approach. (1pt)
    (5) Which approach is taken by TCP? (1pt)

11. (ch37, 7pt) TCP's congestion control mechanism defines how cwnd changes. Consider the following possible changes to cwnd.
    (a) cwnd = 1MSS
    (b) cwnd = (1/2)cwnd
    (c) cwnd = (1/2)cwnd+3MSS
    (d) cwnd = cwnd

(e) cwnd = cwnd+MSS*(MSS/cwnd)

(f) cwnd = cwnd+1MSS

(1) Tell how cwnd is changed when a new ACK is received in the slow start state of a TCP connection. (1pt)

(2) Tell how cwnd is changed when a new ACK is received in the congestion avoidance state of a TCP connection. (1pt)

(3) Tell how cwnd is changed when a duplicate ACK is received in the fast recovery state of a TCP connection. (1pt)

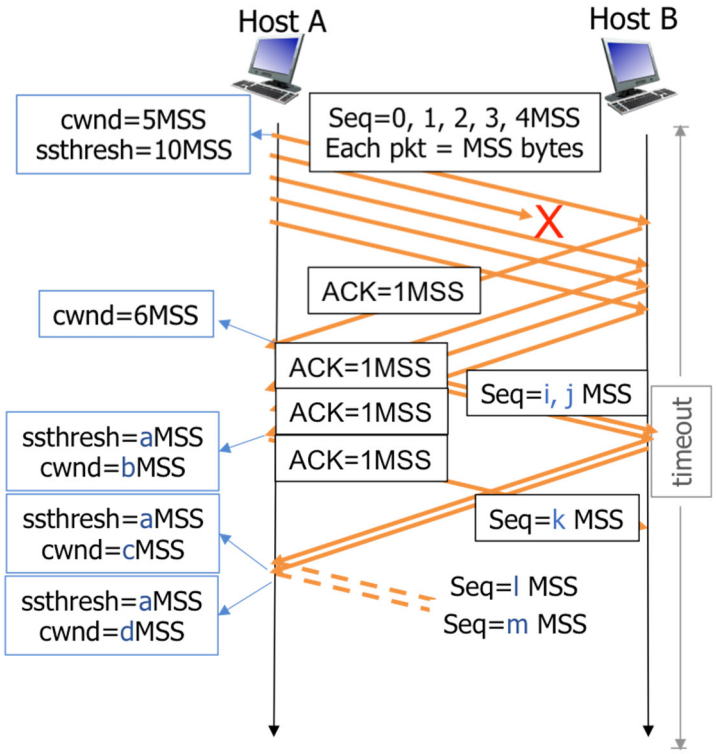
(4) Tell how cwnd is changed when a duplicate ACK is received outside the fast recovery state of a TCP connection. (1pt)

(5) Tell how cwnd is changed when 3 duplicate ACKs are detected in a TCP connection without fast recovery. (1pt)

(6) Tell how cwnd is changed when 3 duplicate ACKs are detected in a TCP connection with fast recovery. (1pt)

(7) Tell how cwnd is changed when a timeout is detected in a TCP connection. (1pt)

12. (ch37, 9pt) The figure below illustrates a part of the data and ACK packet exchange in a TCP connection with fast recovery. Recall how 3 duplicate acks trigger the fast recovery mechanism in TCP and tell the value of a, b, c, d, i, j, k, l, m. Assume that (1) there are always data from above and (2) the packets are always MSS bytes large.



(1) Tell the value of a. (1pt)
(2) Tell the value of b. (1pt)
(3) Tell the value of c. (1pt)
(4) Tell the value of d. (1pt)
(5) Tell the value of i. (1pt)
(6) Tell the value of j. (1pt)
(7) Tell the value of k. (1pt)
(8) Tell the value of l. (1pt)
(9) Tell the value of m. (1pt)

13. (PA, 14pt) Please go on the PA workstation and work under the exam2 directory for this problem set. Create the exam2 directory if you have not yet done so. Grading policy: pts for later problems will be given only when the former ones are completed.

    (1) Develop exam2-p13-1.go such that it connects to the server running on port 12000 and then closes the connection. (2pt)
    (2) Develop exam2-p13-2.go such that it connects to the server running on port 12000, sends "PLAY\n" and then closes the connection. (2pt)
    (3) Develop exam2-p13-3.go such that it connects to the server running on port 12000, sends "PLAY\n", receives a line of message from the server, prints the message on the screen, and then closes the connection. (3pt)

    You should see the response from the server on port 12000. It says it is the "This Number is Closer!" game engine. In this game, a player is asked to figure out the true value (in 1-100) the game engine has in mind. In each round, the player offers two numbers (in 1-100). The game engine compares the distance of the two numbers to the true value and tells which one is closer or whether they are equally close. The game engine responds with "Bingo" when both the numbers are identical to the true value.

    (4) Develop exam2-p13-4.go such that it connects to the server running on port 12000, sends "PLAY\n", receives a line of message from the server, prints the message on the screen. Then, prompts the user for the 1st numbers, sends the number in one line (+'\n' at the end of the number), prompts the user for the 2nd number, sends the number in one line (+'\n' at the end of the number), and then closes the connection. (3pt)
    (5) Develop exam2-p13-5.go such that it connects to the server running on port 12000, sends "PLAY\n", receives a line of message from the server, prints the message on the screen. Then, prompts the user for the 1st numbers, sends the number in one line (+'\n' at the end of the number), prompts the user for the 2nd number, sends the number in one line (+'\n' at the end of the number), receives a line of message from the server, prints the line of message on the screen, and then closes the connection. (3pt)
    (6) Execute exam2-p13-5.go multiple times. Guess a different pair of numbers each time until the game engine responds with "Bingo". Tell what the true value is. (1pt)

14. (PA, 16pt) Please go on the PA workstation and work under the exam2 directory for this problem set. Create the exam2 directory if you have not yet done so. Grading policy: pts for later problems will be given only when the former ones are completed.

    (1) Develop exam2-p14-1.go running on your exam port # such that it allows multiple clients such as your exam2-p13-1.go to connect. (2pt)
    (2) Develop exam2-p14-2.go such that it allows multiple clients such as your exam2-p13-2.go to connect. It allows the client to send a message such as “PLAY\n” and prints the message on screen. (3pt)
    (3) Develop exam2-p14-3.go such that it allows multiple clients such as your exam2-p13-3.go to connect, and allows the client to send a message such as “PLAY\n”. After receiving “PLAY\n”, it sends “Welcome to This Number is Closer! I have a true value in 1-100 for you to figure out. Give me two numbers (in two lines). I will show which one is closer to the true value.\n” back to the client. (4pt)
    (4) Develop exam2-p14-4.go such that it allows multiple clients such as your exam2-p13-4.go to connect, and allows the client to send “PLAY\n”. After receiving “PLAY\n”, it sends “Welcome to This Number is Closer! I have a true value in 1-100 for you to figure out. Give me two numbers (in two lines). I will show which one is closer to the true value.\n” back to the client. Afterwards, allow the client to send the 1st number followed by ‘\n’, print the 1st number on screen, allow the client to send the 2nd number followed by ‘\n’, and print the 2nd number on screen. (3pt)
    (5) Develop exam2-p14-5.go such that it allows multiple clients such as your exam2-p13-5.go to connect, and allows the client to send “PLAY\n”. After receiving “PLAY\n”, it sends “Welcome to This Number is Closer! I have a true value in 1-100 for you to figure out. Give me two numbers (in two lines). I will show which one is closer to the true value.\n” back to the client. Afterwards, allow the client to send the 1st number followed by ‘\n’, print the 1st number on screen, allow the client to send the 2nd number followed by ‘\n’, and print the 2nd number on screen. Finally, send a response telling the number that is closer to the true value, telling “Equally close\n” when the two numbers are equally close, or “Bingo\n” when both numbers are the true value. (4pt)