CS 348: Computer Networks Spring 2025, IIT Dharwad	Mid-semester Exam March 3, 2025	09:30 AM to 11:30 AM Total Marks: 62

Name: ____ **Roll No.** __________________________

Instructions

- This is a question paper-cum-answer sheet, which must be submitted to the invigilator at the end of the exam.
- Answers are to be written exclusively in the space provided after each question.
- Extra sheets may be used for rough work, which need not be submitted.
- No notes, books, cheat-sheet, calculator, etc. are allowed in this exam.

Q1. [*1 mark*] With the Selective-Repeat protocol, it is possible for the sender to receive an ACK for a packet that falls outside of its current window. True or False.

Ans. True. Suppose the sender has a window size of 3 and sends packets 1, 2, 3 at t_0. At $t_1 (> t_0)$, the receiver ACKS 1, 2, 3. At t_2 $(> t_1)$, the sender times out and resends 1, 2, 3. At t_3, the receiver receives the duplicates and re-acknowledges 1, 2, 3. At t_4, the sender receives the ACKs that the receiver sent at and advances its window to 4, 5, 6. At t_5, the sender receives the ACKs 1, 2, 3 the receiver sent at t_2. These ACKs are outside its window.

Q2. [*1 mark*] With GBN, it is possible for the sender to receive an ACK for a packet that falls outside of its current window. True or False.

Ans. True. By essentially the same scenario as in previous question.

Q3. [*1 mark*] A TCP client program can be launched (without errors) before launching the corresponding TCP server. True or False.

Ans. False
Explanation: If you run TCPClient first, then the client will attempt to make a TCP connection with a non-existent server process. A TCP connection will not be made.

Q4. [*1 mark*] A UDP client program can be launched (without errors) before launching the corresponding UDP server. True or False.

Ans. True
Explanation: UDPClient doesn't establish a connection with the server. Thus, everything should work fine if you first run UDPClient, then run UDPServer.

Q5. [*1 mark*] After a TCP handshake, the SMTP handshake is initiated by the ______________.

a. Mail client b. Mail server c. None of the above

Ans. b, as the mail server sends the first message of the SMTP handshake

Q6. [*1 mark*] The Maximum Segment Size (MSS) defines the maximum size of the transport layer _____________.

a. Header
b. Payload
c. Segment (i.e. Header + Payload)
d. None of the above

Ans. b

Q7. [*1 mark*] In a peer-to-peer network, a participating node can:

a. always act only as a client
b. always act only as a server
c. act as a client or a server at any given time
d. act as a client as well as a server *simultaneously*

Ans. d

Q8. [*1 mark*] HTTP response messages never have an empty message body. True or False.

Ans. False

Q9. [*1 mark*] An Internet router ____________ include the transport layer in its network stack.

a. must
b. may
c. does not
d. None of the above

Ans. c

Q10. [*1 mark*] State the HTTP header field name used by a client to convey a persistent/non-persistent connection to the server.

Ans. `Connection`

Q11. [*1 mark*] What is the significance of the `User-agent` header field in the HTTP request message?

Ans. For the server to send different versions of the same object to different types of user agents.

Q12. [*1 mark*] How long does it take a packet of length L to propagate over a link (with ample bandwidth) of distance d, propagation speed s, and transmission rate R?

Ans. $\frac{d}{s}$

Q13. [*2 marks*] Distinguish between Circuit-switched networks vs. Packet-switched networks.

Ans.

Feature	Circuit-Switched Networks	Packet-Switched Networks
Connection Type	Dedicated, continuous connection	Connectionless, dynamic routing of packets
Example	Traditional telephone systems	Internet, email, web browsing
Efficiency	Less efficient for variable data loads	More efficient for bursty, variable traffic
Reliability	High (guaranteed bandwidth)	Variable (depends on protocols like TCP)
Latency	High setup time, constant during the call	Low latency, but depends on congestion
Data Transmission	Continuous flow of data over a fixed path	Data is broken into packets and sent separately
Suitability	Best for voice or constant data streams	Best for variable, bursty data traffic

Q14. [*2 marks*] Derive the formula for the end-to-end delay of sending back-to-back P packets each of length L over N sequential links, each of transmission rate R. Ignore all propagation delays.

Ans. $(N + P - 1) \times \frac{L}{R}$

Explanation: At time $N \times \frac{L}{R}$, the first packet has reached the destination, the second packet is stored in the last router, the third packet is stored in the next-to-last router, etc. At time $N \times \frac{L}{R} + \frac{L}{R}$, the second packet has reached the destination, the third packet is stored in the last router, etc. Continuing with this logic, we see that at time $N \times \frac{L}{R} + (P - 1) \times \frac{L}{R} = (N + P - 1) \times \frac{L}{R}$ all packets have reached the destination.

Q15. [*2 marks*] UDP and TCP use 1s complement for their checksums. Suppose you have the following three 8-bit bytes: `11010011`, `10100110`, `01110100`. What is the 1s complement of the sum of these 8-bit bytes?[1]

[1] Note that although UDP and TCP use 16-bit words in computing the checksum, for this problem you are being asked to consider 8-bit sums.

	1	1	0	1	0	0	1	1	Byte 1
+	1	0	1	0	0	1	1	0	Byte 2
1	0	1	1	1	1	0	0	1	
	0	1	1	1	1	0	1	0	After wraparound
+	0	1	1	1	0	1	0	0	Byte 3
	1	1	1	0	1	1	1	0	
	0	0	0	1	0	0	0	1	One's complement

Ans.

Q16. [*2 marks*] Suppose N packets arrive simultaneously to a link at which no packets are currently being transmitted or queued. Each packet is of length L and the link has transmission rate R. What is the average queuing delay for the N packets?

Ans. $\frac{L(N-1)}{2R}$

Explanation: The queuing delay is 0 for the first transmitted packet, $\frac{L}{R}$ for the second transmitted packet, and generally, $(n-1)\frac{L}{R}$ for the n^{th} transmitted packet. Thus, the average delay for the N packets is:

$$\frac{\left(\frac{L}{R}+\frac{2L}{R}+...+(N-1)\frac{L}{R}\right)}{N} = \frac{L}{RN}(1+2+...+(N-1)) = \frac{L}{RN}\left(\frac{N(N-1)}{2}\right) = \frac{L(N-1)}{2R}$$

Q17. [*2 marks*] Suppose Bob joins a BitTorrent torrent, but he does not want to upload any data to any other peers (so called free-riding). Bob claims that he can receive a complete copy of the file that is shared by the swarm. Is Bob's claim possible? Why or why not?

Ans. Yes

Explanation: His first claim is possible, as long as there are enough peers staying in the swarm for a long enough time. Bob can always receive data through optimistic unchoking by other peers.

Q18. [*2 marks*] State and explain the HTTP header fields used by client and server for web caching.

Ans.

Client-triggered: `If-modified-since`

Server-triggered: `Cache-control`

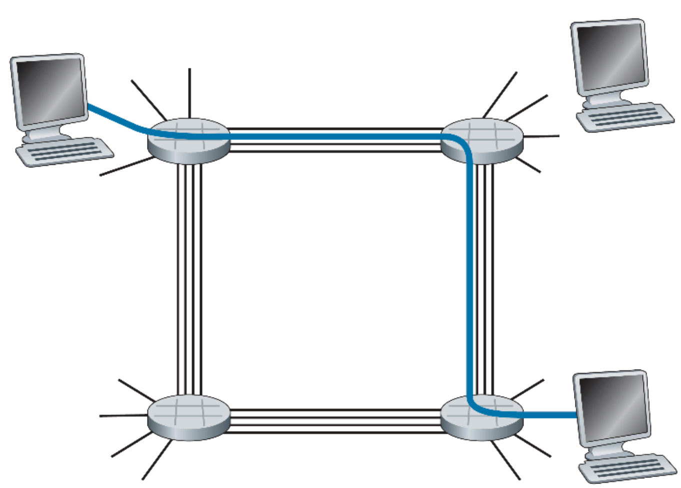
Q19. [*3 marks*] Consider sending real-time voice from Host A to Host B over a packet-switched network (VoIP). Host A converts analog voice to a digital 64 kbps bit stream on the fly. Host A then groups the bits into 56-byte packets. There is one link between Hosts A and B; its transmission rate is 10 Mbps and its propagation delay is 10 msec. As soon as Host A gathers a packet, it sends it to Host B. As soon as Host B receives an entire packet, it converts the packet's bits to an analog signal. How much time elapses on an average from the time a bit is created (from the original analog signal at Host A) until the bit is decoded (as part of the analog signal at Host B)?

Ans. Consider the first bit in a packet. Before this bit can be transmitted, all of the bits in the packet must be generated. This requires $\frac{56\times8}{64\times10^3} = 7$ milliseconds.

The time required to transmit the packet is $\frac{56\times8}{10\times10^6} = 44.8$ microseconds.

Propagation delay = 10 milliseconds.

The delay until decoding is $7\times10^{-3} + 44.8\times10^{-6} + 10\times10^{-3} = 17.0448$ milliseconds.



Q20. [*3 marks*] Consider the adjoining circuit-switched network. There are four circuits on each link (i.e. pair of neighboring routers). Label the four switches A, B, C, and D, going in the clockwise direction.

a. What is the maximum number of simultaneous connections that can be in progress at any one time in this network?
b. Suppose that all connections are between switches A and C. What is the maximum number of simultaneous connections that can be in progress?
c. Suppose we want to make four connections between switches A and C, and another four connections between switches B and D. Can we route these calls through the four links to accommodate all eight connections?

Ans. (a) 16, (b) 8, (c) Yes
<u>*Explanation:*</u>
a) Between the switch in the upper left and the switch in the upper right we can have 4 connections. Similarly, we can have four connections between each of the 3 other pairs of adjacent switches. Thus, this network can support up to 16 connections.
b) We can have 4 connections passing through the switch in the upper-right-hand corner and another 4 connections passing through the switch in the lower-left-hand corner, giving a total of 8 connections.
c) Yes. For the connections between A and C, we route two connections through B and two connections through D. For the connections between B and D, we route two connections through A and two connections through C. In this manner, there are at most 4 connections passing through any link.

Q21. [*3 marks*] A sender and receiver communicate over a channel that may corrupt data but cannot lose data. The corresponding finite state machines (FSMs) of the sender and the receiver are as follows. Show that this can lead the sender and receiver to enter into a deadlock state, where each is waiting for an event that will never occur.

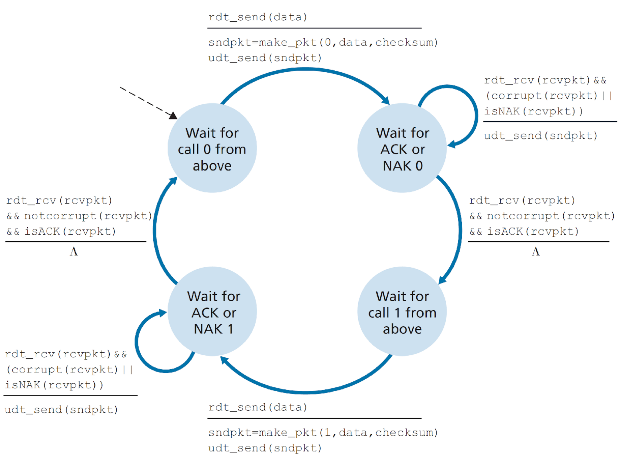


Figure 1: Sender's FSM

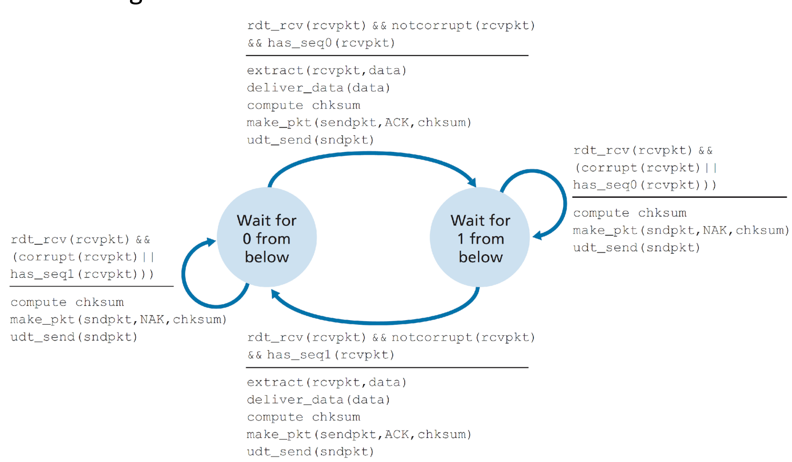


Figure 2: Receiver's FSM

Ans. Suppose the sender is in state “Wait for call 1 from above” and the receiver (the receiver shown in the homework problem) is in state “Wait for 1 from below.” The sender sends a packet with sequence number 1, and transitions to “Wait for ACK or NAK 1,” waiting for an ACK or NAK. Suppose now the receiver receives the packet with sequence number 1 correctly, sends an ACK, and transitions to state “Wait for 0 from below,” waiting for a data packet with sequence number 0. However, the ACK is corrupted. When the rdt2.1 sender gets the corrupted ACK, it resends the packet with sequence number 1. However, the receiver is waiting for a packet with sequence number 0 and always sends a NAK when it doesn't get a packet with sequence number 0. Hence the sender will always be sending a packet with sequence number 1, and the receiver will always be NAKing that packet. Neither will progress forward from that state.

Q22. [*3.5 marks*] Host A and B are communicating over a TCP connection, and Host B has already received from A all bytes up through byte 126. Suppose Host A then sends two segments to Host B back-to-back. The first and second segments contain 80 and 40 bytes of data, respectively. In the first segment, the sequence number is 127, the source port number is 302, and the destination port number is 80. Host B sends an acknowledgment whenever it receives a segment from Host A.

a. In the second segment sent from Host A to B, what are the sequence number, source port number, and destination port number?
b. If the first segment arrives before the second segment, in the acknowledgment of the first arriving segment, what is the acknowledgment number, the source port number, and the destination port number?
c. If the second segment arrives before the first segment, in the acknowledgment of the first arriving segment, what is the acknowledgment number?

Ans.
a) [0.5 mark×3] In the second segment from Host A to B, the sequence number is 207, source port number is 302 and destination port number is 80.
b) [0.5 mark×3] If the first segment arrives before the second, in the acknowledgement of the first arriving segment, the acknowledgement number is 207, the source port number is 80 and the destination port number is 302.
c) [0.5 mark] If the second segment arrives before the first segment, in the acknowledgement of the first arriving segment, the acknowledgement number is 127, indicating that it is still waiting for bytes 127 and onwards.

Q23. [*4 marks*] Consider sending a packet from a source host to a destination host over a fixed route. List the delay components in the end-to-end delay. Which of these delays are constant and which are variable?

Ans.
[0.5×4 marks] The delay components are processing delays, transmission delays, propagation delays, and queuing delays.
[0.5×4 marks] All of these delays are fixed, except for the queuing delays, which are variable.

Q24. [*4 marks*] Consider sending a large file of F bits from Host A to Host B. There are three links (and two switches) between A and B, and the links are uncongested (that is, no queuing delays). Host A segments the file into segments of S bits each and adds 80 bits of header to each segment, forming packets of $L = 80 + S$ bits. Each link has a transmission rate of R bps. Find the value of S that minimizes the delay of moving the file from Host A to Host B. Disregard propagation delay.

Ans. There are $\frac{F}{S}$ packets. Each packet is $S + 80$ bits long. Time at which the last packet is received at the first router is $\frac{S+80}{R} \times \frac{F}{S}$ sec. At this time, the first $\frac{F}{S} - 2$ packets are at the destination, and the $\frac{F}{S} - 1$ packet is at the second router. The last packet must then be transmitted by the first router and the second router, with each transmission taking $\frac{S+80}{R}$ sec. Thus, delay in sending the whole file is $delay = \frac{S+80}{R} \times \left(\frac{F}{S} + 2\right)$. To calculate the value of S which leads to the minimum delay,

$$\frac{d}{dS}delay = 0 \Rightarrow S = \sqrt{40F}$$

Q25. [*4 marks*] Consider transferring an enormous file of L bytes from Host A to Host B. Assume an MSS of 536 bytes.
a. What is the maximum value of L such that TCP sequence numbers are not exhausted?
b. For the L you obtain in (a), find how long it takes to transmit the file. Assume that a total of 66 bytes of transport, network, and data-link header are added to each segment before the resulting packet is sent out over a 155 Mbps link. Ignore flow control and congestion control so A can pump out the segments back-to-back and continuously.

Ans. There are 2^{32} possible sequence numbers.
(a) The sequence number does not increment by one with each segment. Rather, it increments by the number of bytes of data sent. So, the size of the MSS is irrelevant -- the maximum size file that can be sent from A to B is simply the number of bytes representable by $2^{32} \approx 4.19\ GB$.
(b) The number of segments is $\lceil \frac{2^{32}}{536} \rceil = 8012999$. 66 bytes of header get added to each segment giving a total of 528,857,934 bytes of header. The total number of bytes transmitted is $2^{32} + 528,857,934 = 4.824 \times 10^{9}$ bytes, which would take 249 seconds to transmit the file over a 155~Mbps link.

Q26. [*4.5 marks*] Suppose that five consecutive measured `SampleRTT` values are 106 ms, 120 ms, and 140 ms. Compute the `EstimatedRTT` after each of these `SampleRTT` values is obtained, using a value of $\alpha = 0.125$ and assuming that the value of `EstimatedRTT` was 100 ms just before the first of these five samples were obtained. Compute also the `DevRTT` after each sample is obtained, assuming a value of $\beta = 0.25$ and assuming the value of `DevRTT` was 5 ms just before the first of these five samples was obtained. Last, compute the TCP `TimeoutInterval` after each of these samples is obtained.

Ans.
DevRTT = (1- beta) * DevRTT + beta * | SampleRTT - EstimatedRTT |
EstimatedRTT = (1-alpha) * EstimatedRTT + alpha * SampleRTT
TimeoutInterval = EstimatedRTT + 4 * DevRTT

<u>After obtaining first SampleRTT 106ms:</u>
[0.5 mark] DevRTT = 0.75*5 + 0.25 * | 106 - 100 | = 5.25ms
[0.5 mark] EstimatedRTT = 0.875 * 100 + 0.125 * 106 = 100.75 ms
[0.5 mark] TimeoutInterval = 100.75+4*5.25 = 121.75 ms

<u>After obtaining 120ms:</u>
[0.5 mark] DevRTT = 0.75*5.25 + 0.25 * | 120 – 100.75 | = 8.75 ms
[0.5 mark] EstimatedRTT = 0.875 * 100.75 + 0.125 * 120 = 103.16 ms
[0.5 mark] TimeoutInterval = 103.16+4*8.75 = 138.16 ms

<u>After obtaining 140ms:</u>
[0.5 mark] DevRTT = 0.75*8.75 + 0.25 * | 140 – 103.16 | = 15.77 ms
[0.5 mark] EstimatedRTT = 0.875 * 103.16 + 0.125 * 140 = 107.76 ms
[0.5 mark] TimeoutInterval = 107.76+4*15.77 = 170.84 ms

Q27. [*8 marks*] Suppose within your Web browser you click on a link to obtain a Web page. The IP address for the associated URL is not cached in your local host, so a DNS lookup is necessary to obtain the IP address. Suppose that n DNS servers are visited before your host receives the IP address from DNS; the successive visits incur an RTT of $RTT_1, \ldots, RTT_n$. Further suppose the HTML file (consisting of a small amount of HTML text) references eight very small objects on the same server. Let RTT_0 denote the RTT between the local host and the server containing the object. Neglecting transmission times, how much time elapses from when the client clicks on the link until the client receives the complete webpage (including all embedded objects) considering:

a. Non-persistent HTTP with no parallel TCP connections and no pipelining?
b. Non-persistent HTTP with 6 parallel connections and no pipelining?
c. Persistent HTTP with pipelining but no parallel connections?
d. Persistent HTTP with 7 parallel connections but no pipelining?

Ans. The total amount of time to get the IP address is $RTT_1 + RTT_2 + \ldots + RTT_n$. Once the IP address is known, RTT_0 elapses to set up the TCP connection and another RTT_0 elapses to request and receive the base HTML object. Thus, a total of $t_{default} = \sum_i^n RTT_i + 2RTT_0$.

a. In addition to $t_{default}$, for each referenced object, $2RTT_0$ will be required, i.e. $\sum_i^n RTT_i + 18RTT_0$.

b. In addition to $t_{default}$, for every six referenced objects, $2RTT_0$ will be required, i.e. $\sum_i^n RTT_i + 6RTT_0$.

c. In addition to $t_{default}$, all referenced objects will be requested back-to-back, requiring RTT_0, i.e. $\sum_i^n RTT_i + 3RTT_0$.

d. In addition to $t_{default}$, six additional connections will be launched each requiring $2RTT_0$, while during this time the original connection can fetch two objects (each requiring only 1 RTT_0), thereby a total of $\sum_{i}^{n} RTT_i + 4RTT_0$.