1. For each of the following pairs of technical terms, define each term, and clarify the key difference(s) between the two terms. Be clear and concise. If in doubt about your definition, feel free to supplement with a relevant example.

   (a) "IPv4 address" and "MAC Address" **(1 point)**

   IPv4 address: A 32-bit logical address assigned to a device on a network using the Internet Protocol (e.g., 192.168.1.1). It operates at the network layer (Layer 3) of the OSI model and is used for routing data across different networks. IPv4 addresses are hierarchical, configurable, and can change (e.g., via DHCP).

   MAC Address: A 48-bit physical address hard-coded into a device's network interface card (e.g., 00:1A:2B:3C:4D:5E). It operates at the data link layer (Layer 2) of the OSI model and ensures data delivery within a local network segment. MAC addresses are flat, static, and unique to each hardware interface.

   (b) "DHCP" and "NAT" **(1 point)**

   DHCP: A network protocol that automatically assigns IP addresses and other network configuration parameters (e.g., subnet mask, default gateway, DNS server) to devices joining a network. It operates at the application layer (Layer 7) and simplifies IP management by leasing addresses dynamically (e.g., assigning 192.168.1.10 to a new device).

   NAT: A method used to translate private IP addresses (e.g., 192.168.1.10) on a local network to a public IP address (e.g., 203.0.113.5) when communicating with the internet. It operates at the network layer (Layer 3) and conserves IPv4 addresses by allowing multiple devices to share a single public IP.

   (c) "Congestion control" and "Flow control" **(1 point)**

   Congestion Control: A mechanism to prevent network overload by regulating the amount of data sent into the network. It operates at the transport layer (Layer 4) and ensures that routers or links do not become congested due to excessive traffic. Congestion control adjusts transmission rates dynamically using algorithms like TCP's AIMD (Additive Increase/Multiplicative Decrease) or slow start.

   Flow Control: A mechanism to ensure a sender does not overwhelm a receiver with data. It operates at the transport layer (Layer 4) and manages the data flow between two endpoints (e.g., client and server). Flow control uses methods like sliding window protocols or receiver window size (advertised in TCP headers) to limit the sender's transmission rate based on the receiver's buffer capacity.

2. With distance vector routing, router A has the following routing table at time instant t:

cost to	A/thru	B/thru	C/thru	D/thru	E/thru	F/thru	G/thru
from A	0/A	3/A	not known	8/A	not know	not know	not know

At time t, router A receives the following routing update from router B:

cost	to A	to B	to C	to D	to E	to F	to G
from B	3	0	7	2	3	7	7

the following routing update from router D:

cost	to A	to B	to C	to D	to E	to F	to G
from D	5	2	4	0	1	7	7

Please write down router A's routing table after receiving these two routing updates, respectively. **(4 points)**

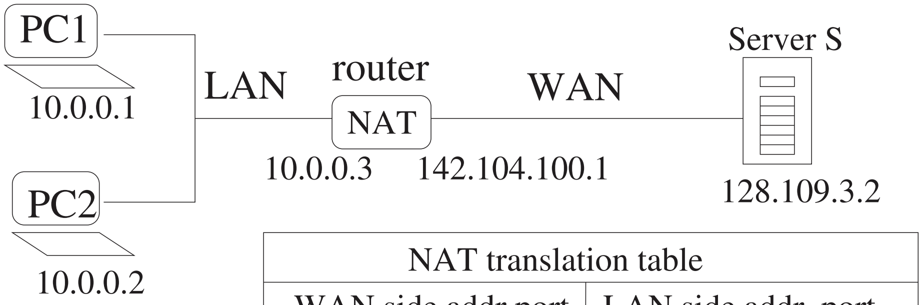
After router B's update

cost to	A/thru	B/thru	C/thru	D/thru	E/thru	F/thru	G/thru
from A	0/A	3/A	10/B	5/B	6/B	10/B	10/B

After router D's update

cost to	A/thru	B/thru	C/thru	D/thru	E/thru	F/thru	G/thru
from A	0/A	3/A	9/D	5/B	6/B	10/B	10/B

3.



NAT translation table	
WAN side addr,port	LAN side addr, port
142,104.100.1, 5001 142.104.100.1, 5002	10.0.0.1, 5002 10.0.0.2, 5001 ......

As shown in the figure, two PCs are in the same local area network (LAN), and the router connecting the LAN and the WAN performs the network address translation (NAT). PC1 establishes a TCP connection with Server S (connection 1), with PC1's TCP port number 5002 and Server S's TCP port number 8000; PC2 establishes a TCP connection with Server S, with PC2's TCP port number 5001 and Server S's TCP port number 25 (connection 2). Write down the source/destination IP addresses and TCP port numbers of the following packets:
packet 1: belong to connection 1, source is Server S, destination is PC1, in the LAN;
packet 2: belong to connection 1, source is Server S, destination is PC1 in the WAN;
packet 3: belong to connection 2, source is PC2, destination is Server S, in the LAN;
packet 4: belong to connection 2, source is PC2, destination is Server S, in the WAN. **(4 points)**

packet No.	source IP	source port	destination IP	destination port
1	128.109.3.2	8000	10.0.0.1	5002
2	128.109.3.2	8000	142.104.100.1	5001
3	10.0.0.2	5001	128.109.3.2	25
4	142.104.100.1	5002	128.109.3.2	25

4. Suppose you want to send a file of size 130,000 bytes over a 2 Mbps link using TCP. The maximum segment size (MSS), which represents the size of TCP payload, is 1,000 bytes. Two-way propagation delay between the source and the destination is 10 [msec]. TCP Slow start threshold is 32 MSS, initial congestion window (cwnd) is 1 MSS, and the initial receiver window size is 200 MSS.

   How long will it take to transmit the given file, from the TCP connection establishment to the termination, assuming there is no packet loss throughout the transmission? **(7 points)**

   Ans:

   1st RTT - connection established

   2nd RTT-window: 1MSS => transmitted bytes: 1,000

   3rd RTT - window: 2MSS => transmitted bytes: $1,000 + 2,000 = 3,000$

   4th RTT - window: 4MSS => transmitted bytes: $3,000 + 4,000 = 7,000$

   5th RTT - window: 8 MSS => transmitted bytes: $7,000 + 8,000 = 15,000$

   6th RTT - window: 16MSS => transmitted bytes: $15,000 + 15,000 = 31,000$

   7th RTT - window: 32 MSS => transmitted bytes: $31,000 + 32,000 = 63,000$

   8th RTT - window: 33 MSS => transmitted bytes: $63,000 + 33,000 = 96,000$

   9th RTT - window: 34 MSS => transmitted bytes: $96,000 + 34,000 = 130,000$

   Delay: $9 * 2$-way propagation $+(1 + 2 + 4 + 8 + 16 + 32 + 64 + 128)^*$ packet-transmissions $= = 9 * 0.01$ [sec] $+130 * 1,020 * 8$ [bits] $/2,000,000$ [bit/sec] =

   $$= 0.09 + 0.0.5304[\text{sec}] = 0.6204[\text{sec}]$$

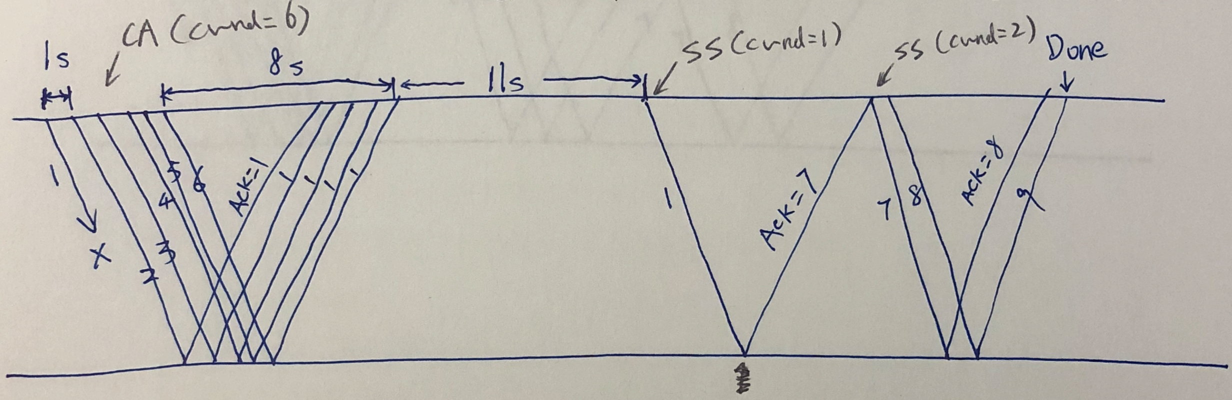
   We could possibly add another 1.5 RTT for the (2-way) closing of the connection.

   So the final duration is $0.6204 + 1.5 * 0.01 = 0.6354$s.

5. Consider a TCP connection that, at time 0, is in the congestion avoidance phase with a window size equal to 6 MSS, and then sends packets {1, 2, 3, .., 6}. (The connection has sent packet before that were properly acknowledged. The packet transmission time of a single packet is $P_{\text{transmission}} = 1$ sec, while RTT = 8 sec and timeout = 24 sec.) The sender does *not* implement fast retransmit and fast recovery, and after every timeout it restarts in slow start.

   (a) Now, assume packets 1 is lost. Draw a timing diagram that shows the transmission of the first 8 packets by completing the figure below. Indicate the significant events, such as start of slow start (SS), or of congestion avoidance (CA), and the relevant congestion window sizes.

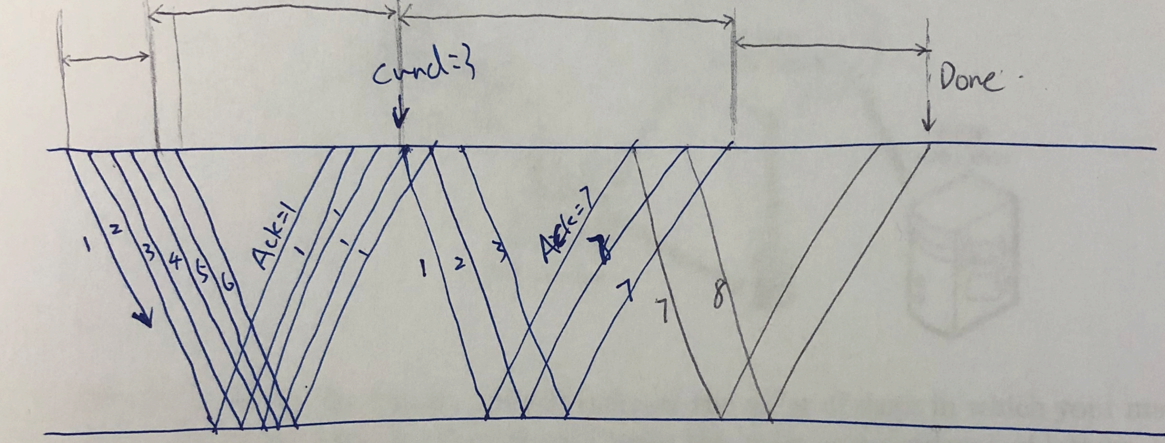
   Also, calculate how long, overall, it takes to transmit those 8 packets. **(5 points)**



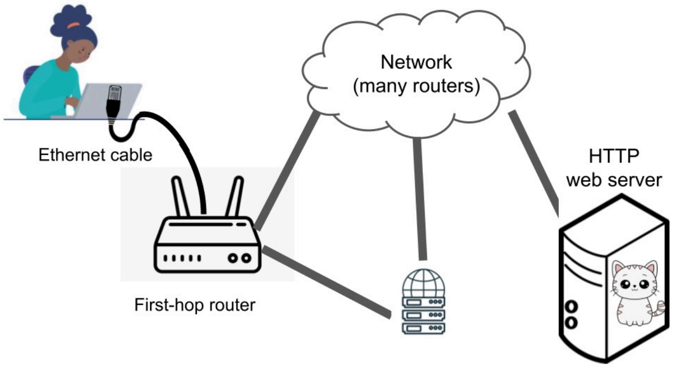
Assuming we're using Selective Repeat to reTx only lost packets, the total time is

24 + 8 + 1 + 8 = 41s

(b) Repeat the previous problem assuming that the host now uses fast retransmit and fast recovery. **(5 points)**



6. What happens when you type a URL in your Web browser and press “enter”?



Please (1) fill in the blanks; and (2) indicate the order of steps in which your machine will send or receive these packets. (You should write the order as a sequence of letters, e.g., BDECA...; all letters should be listed exactly once.)

Hint: terms that might be useful in filling in the blanks:

- *DNS resolver*
- *Your machine (client)*
- *Target web server*
- *First-hop router (from your machine/the client)*

(A) A TCP segment with the SYN flag and the ACK flag set; sent by **server** to **client**. **(1 point)**

(B) A TCP segment with the SYN flag set (ACK flag not set) sent by **client** to **server**. **(1 point)**

(C) A TCP segment with the ACK flag set (SYN flag not set) sent by **client** to **server**. **(1 point)**

(D) HTTP request(s) over an established TCP connection, asking for the content of the Web page sent by **client** to **server**. **(1 point)**

(E) HTTP response(s) over an established TCP connection, containing the contents of the Web page sent by **server** to **client**. **(1 point)**

(F) A DNS request sent by **client** to **dns resolver**, requesting the **IP** (IP / Ethernet: choose one) address of **server**. **(2 points)**

(G) An ARP request sent by your machine to Ethernet broadcast, requesting the **Ethernet** (IP / Ethernet - choose one) address of **first router**. **(1 point)**

(H) An ARP response sent by **first router** to your machine. **(0.5 points)**

(I) A DNS response sent by **dns resolver** to **client**, providing the **ip** (IP / Ethernet - choose one) address of **server**. **(2 points)**

Order of steps (your answer here): **(1.5 points)**

In general, the client sends DNS requests to a DNS server, which might require knowing the router's MAC. So before sending the DNS request (F), the client might need to ARP for the router's MAC (G and H). Because the DNS request is sent via the network, which goes through the router.

But maybe the DNS resolver is local. So we allow two answers:

1) G → H → F → I → B → A → C → D → E

or

2) F → I → G → H → B → A → C → D → E

— END —