

Name: \_\_\_\_\_

Entry Number: \_\_\_\_\_

## Minor Exam – Solution

### COL334/672: Computer Networks

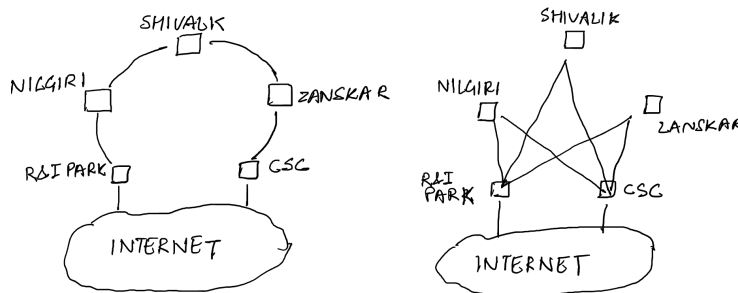
### Sem I, 2025-26

There are 16 questions and 17 pages in this exam booklet (including this page). There are **75 total points**, and you have **120 minutes** to answer the questions.

- **Feel free to think outside the box but write inside the box**
- **Write concise answers**
- **Do not start the exam until instructed to do so**

## I Link Layer

**1. [3 points]:** Consider the two design approaches below for connecting IITD hostels to the Internet using optical fiber. Compare these designs in terms of their cost, resilience, and capacity? Explain briefly.



**Solution:**

Criteria	First Design	Second Design
Cost	Cheaper since it has fewer links	Expensive as it requires more links
Resilience	Moderately resilient: if one link fails, connectivity can still be maintained, but if two simultaneous links fail, the corresponding nodes are cut off.	More resilient: multiple alternative paths exist. However, hostels still have only 2 links – if both fail, the hostel can be cut off.
Capacity	Bandwidth is shared across fewer links, so congestion can occur under heavy load.	Higher capacity since multiple independent paths exist between hostel and Internet.

**2. [4 points]:** Consider a CRC generator polynomial:  $G(x) = x^{16} + x^{12} + x^5 + 1$ . A burst error of length  $L$  is an error pattern where the corrupted bits are confined to exactly  $L$  consecutive positions, with errors in the first and last of those positions. Prove that any burst error of length  $L \leq 16$  is always detected by this CRC polynomial.

**Solution:**

- A.**  $G(x) = x^{16} + x^{12} + x^5 + 1$ , so  $\deg G = 16$  and  $G(0) = 1$ . Thus  $\gcd(G(x), x) = 1$ .
- B.** Any burst of length  $L \leq 16$  can be written  $E(x) = x^k B(x)$ , with  $\deg B = L - 1 \leq 15$ ,  $B(0) = 1$ , and leading coefficient 1.
- C.** If  $G(x) \mid E(x)$  then  $G(x) \mid B(x)$  (since  $G$  and  $x$  are coprime).
- D.** But  $\deg B \leq 15 < 16 = \deg G$ , so  $B(x)$  cannot be a multiple of  $G(x)$ .
- E.** Hence  $G(x) \nmid E(x)$  and every burst of length  $L \leq 16$  is detected.

**3. [4 points]:** Recall that Manchester encoding uses two signals to represent a single bit. Alternatively, this can be viewed as a two-step process:

**Step 1.** Bit encoding:  $1 \rightarrow 10, 0 \rightarrow 01$ .

**Step 2.** Signal encoding: The encoded bits are transmitted using NRZ scheme ( $0 \rightarrow \text{low}, 1 \rightarrow \text{high}$ )

Let's call this scheme as 1B/2B encoding as 1 bit is encoded using 2 bits. Given this context, answer the following questions:

- A.** Design a 2B/3B encoding scheme such that a signal transition occurs at least once every 4-bit times. Explain both the bit-encoding and signal-encoding steps.
- B.** For a general  $nB/n + 1B$  encoding scheme, discuss one advantage and one disadvantage of using a large value of  $n$  for such encoding.

**Solution:**

**A. Bit encoding:** 3-bit codewords should be written such that in every 4 bits there is a transition. An example of bit encoding is given below:  $00 \rightarrow 010, 01 \rightarrow 011, 10 \rightarrow 100, 11 \rightarrow 101$

**B. Signal encoding:** Using NRZ:  $0V \rightarrow \text{low}, 1V \rightarrow \text{high}$

**C. Disadvantage:** You have to wait for  $n$  bits to send, so latency increases when the number of bits  $< n$  or when lookup takes longer time.

**D. Advantage:** Only one extra bit is added per  $n$  bits, so overhead is smaller and you can transmit more data per second.

**4. [4 points]:** Answer the following questions:

- A. Recall that the minimum frame size on 10 Mbps Ethernet is 64 bytes. Assume a new version of Ethernet called *Fast Ethernet* that uses the same medium access control as Ethernet. If we want to keep the minimum frame size the same, what should a network designer ensure for correct MAC operation?
- B. Consider two nodes, A and B, that use the slotted ALOHA protocol to contend for a channel. Suppose node A's transmission probability is  $p$ , while node B's transmission probability is  $2p$ . Provide a formula for A's average throughput. In addition, find  $p$  such that A's throughput is thrice of B's throughput.

**Solution:**

- (A) Minimum frame = 64 bytes = 512 bits. Transmission time  $T_{trans} = N/R$ . For CSMA/CD:  $T_{trans} \geq 2\tau_{max}$ , where  $\tau_{max} = d/s$  ( $d$ =max cable length,  $s$ =signal speed). If rate increases by factor  $k$ , then  $T'_{trans} = T_{trans}/k$ , so  $\tau_{max}$  must shrink by factor  $k$ . Since the signal propagation speed  $s$  is a property of the physical medium and cannot be changed significantly between different media, the only practical option is to reduce the maximum cable length proportionally as the bit rate increases. In this case for a 10× increase in bit rate the maximum allowed length must be 10× smaller.
- (B) In slotted ALOHA, A succeeds if it transmits while B does not. Thus  $S_A = p(1 - 2p)$  and  $S_B = 2p(1 - p)$ . If we want  $3S_A = S_B$ , then  $3 \cdot (1 - 2p) = (2p(1 - p))$ . For  $p > 0$ , dividing gives  $3 - 6p = (2 - 2p)$ , which simplifies to  $p = 0.25$ .

**5. [5 points]:** Consider a wireless network with three nodes: A, B, and C. A and B can hear each other, B and C can hear each other.

- A. Explain why A and C cannot transmit simultaneously using CSMA/CD.
- B. Explain why this issue does not arise in wired Ethernet networks.
- C. Sketch a MAC protocol that solves this problem in the wireless scenario.

**Solution:**

- A. Wireless transmission (CSMA/CD):** You cannot transmit and sense at the same time, which causes the hidden terminal problem.
- B. Wired medium (CSMA/CD):** You can transmit and sense at the same time. Everyone can listen to everyone else, so collisions can be detected quickly.
- C. CSMA/CA with RTS/CTS:** The transmitting device first sends a Request to Send (RTS) signal. The receiving device responds with a Clear to Send (CTS) signal, granting permission to send. This helps avoid hidden terminal collisions in wireless communication.

**6. [3 points]:** The following character encoding is used in a data link protocol:

A: 01000111      B: 11100011      FLAG: 01111110      ESC: 11100000

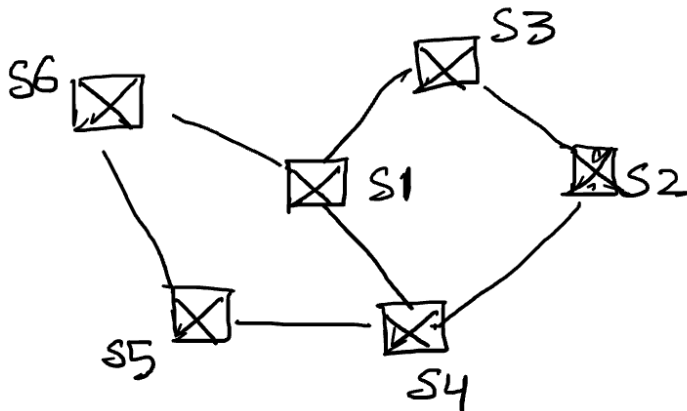
Show the bit sequence transmitted for the four-character frame A B ESC FLAG when each of the following framing methods is used:

- A. Byte count
- B. Flag bytes (or sentinel approach) with byte stuffing
- C. Flag bytes (or sentinel approach) with bit stuffing

Feel free to report Part A and Part B in terms of symbols, while Part C should be shown in binary bits.

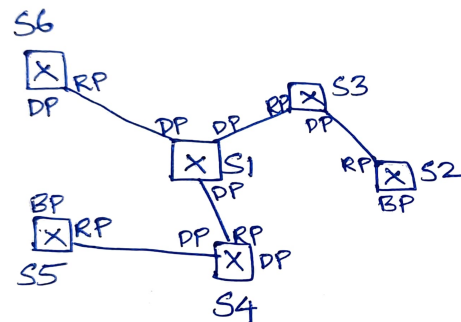
**Solution:**

- A. **Byte count format:** [count] [A] [B] [ESC] [FLAG] → 00000100 [A] [B] [ESC] [FLAG] → 4 [A] [B] [ESC] [FLAG].
- B. **Flag bytes with byte stuffing:** Frame is delimited by [FLAG], and if a data byte equals FLAG or ESC, prefix it with [ESC]. Sequence: [FLAG] [A] [B] [ESC] [ESC] [ESC] [FLAG] [FLAG].
- C. **Flag byte with bit stuffing:** Whenever there are 5 consecutive 1s in the data, insert a 0 immediately after. Frame is delimited by [FLAG]. Sequence (binary): [FLAG] 01000111111010001111100000011111010 [FLAG]

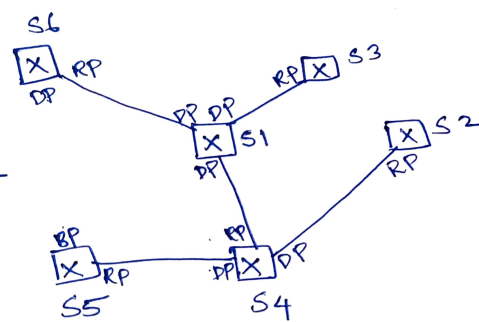


7. [4 points]: For the given Layer-2 network topology, draw the spanning tree generated by the Spanning Tree Protocol (STP). Clearly indicate the root port, designated ports, and blocked ports. What changes occur in the spanning tree when the link between S2 and S3 goes down? In particular, show the new spanning tree and explain how it is constructed.

**Solution:**



Initial spanning tree



Spanning tree after S2-S3 goes down

- When the link between S2 and S3 goes down, S2 detects the failure immediately. This causes S2 to invalidate the information associated with its root port (the port connected to S3). At this point, S2 re-evaluates its available ports to find a new path to the root (S1). The port on S2 connected to S4 was previously in a blocked state, but it has been receiving configuration messages from S4 all along, even while blocked. These messages inform S2 of an alternate path to the root S1 via S4-S1. Since this is now the only viable path, S2 selects its port to S4 as the new root port.
- S3 also detects the link down but, since its port to S2 was a designated port, it simply takes the port out of service without further changes to the tree on its side. No other switches are impacted

## II Network Layer

8. [3 points]: Explain the difference between routing, forwarding, and switching.

**Solution:**

- A. Routing:** Process of computing the best path for a data packet to travel from a source to a destination across the network.
- B. Forwarding:** While routing selects the best path, forwarding is the process of moving a packet from an input interface to an output interface based on the forwarding table.
- C. Switching:** The mechanism inside devices (routers/switches) to transfer the packet from an input port to the appropriate output port efficiently.

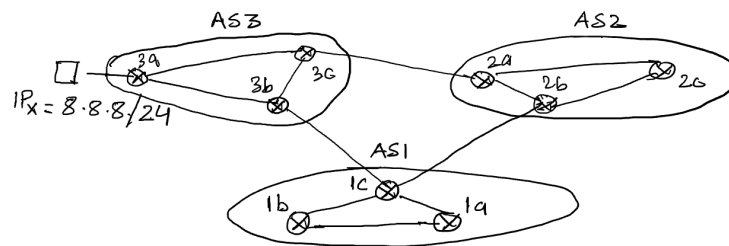
**9. [3 points]:** Explain how large file transfers could degrade the latency observed by both a gaming application and small file transfers. Suggest a scheduling mechanism that could mitigate this problem.

**Solution:** Large file transfers consume significant bandwidth and buffer space in routers/switches. This leads to queuing delays, which increase the latency experienced by small packets such as those from gaming applications or small file transfers.

**Solution: Scheduling Mechanisms:**

- **Weighted Fair Queueing (WFQ):** Allocates bandwidth fairly among flows, ensuring that large file transfers do not dominate.
- **Priority Queueing:** Gives higher priority to latency-sensitive traffic (e.g., gaming packets) over bulk data transfers.
- etc.

**10. [6 points]:** Consider the network topology below.  $AS3$  is the origin AS for IP prefix  $IP_x = 8.8.8.8/24$ . Assume all routers are BGP routers. Answer the following questions:



- Assuming synchronous updates, list all the announcements for  $IP_x$  received at BGP routers  $1c$  and  $1b$ . Be sure to include both the AS path and the next hop.
- Case 1: Suppose  $AS1$  wants to route traffic to  $IP_x$  via  $AS2$ . How can it achieve this?
- Case 2: Suppose  $AS2$  wants the traffic from  $AS1$  to  $IP_x$  to pass through its network (e.g.,  $AS2$  is a malicious AS trying to surveil  $AS1$ 's traffic). Design a mechanism to accomplish this using only BGP. (Hint: Consider the IP lookup process.)

**Solution:**

a. For 1c

- AS3  $IP_x$  ; 3b
- AS2 AS3  $IP_x$  ; 2b

For 1b

- AS3  $IP_x$  ; 3b

*FYI: In BGP, each router selects a single best route for a given prefix and propagates only this best route to its internal (iBGP) peers. Internal routers themselves do not originate or modify external routes; they only forward the best route they learn.*

- b. To prefer AS2, set a higher LOCAL\_PREF for routes learned from AS2. Since LOCAL\_PREF is an attribute used inside an AS to choose the preferred exit point among multiple options, this policy makes AS1 prefer sending traffic to  $IP_x$  via AS2, even if the AS-path is longer.
- c. AS2 Employs sub-prefix hijacking. AS2 maliciously advertises the following two more specific IPs of  $IP_x$ , which in IP lookup table are preferred due to longest prefix match, regardless of AS-path length.
- 8.8.8.0/25
  - 8.8.8.128/25

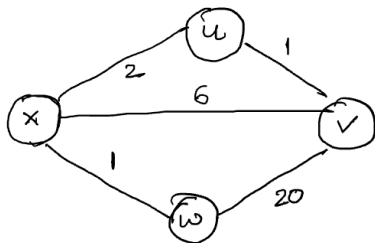
**11. [3 points]:** Recall that in interdomain routing, a link between two ASes can represent one of three business relationships: customer-provider, provider-customer, or peer-peer. As a network operator, you must configure BGP export policy – i.e., decide which neighbours (peers, customer, providers) should be forwarded a route advertisement – while ensuring that the policy respects the underlying economic relationships. For each of the following cases, specify the correct export policy:

- route was learned from a customer
- route was learned from a provider
- route was learned from a peer

**Solution:**

- Forward to customers, providers and peers
- Forward to customers only
- Forward to customers only

**12. [8 points]:** Consider the following network topology using distance vector routing with poisoned reverse, i.e., if A routes to X through neighbor B, A does not share its distance to X with B. Answer the following questions:



- Give  $x$ 's distance vector announcements to  $u$ .
- Give a link-cost change for either  $c(v,w)$  or  $c(v,u)$  such that  $v$  will *not* inform its neighbors of a new minimum cost-path to any destination to its neighbors.
- Suppose that  $c(x,w)$  increases to 40. Will there be a count-to-infinity problem? If no, explain why. If yes, how many messages will be exchanged between  $x$  and  $v$  before the network stabilizes again?
- How do you modify  $c(x,y)$  such that there is no count-to-infinity problem at all if  $c(x,w)$  changes from 1 to 40?

**Solution:**

**A.**  $\{x: 0, v: 6, w: 1\}$  at  $t = 0$  and  $\{x: 0, w: 1\}$  at  $t = 1$

**B.** A node sends an update only when its own distance vector changes.

\* If we change  $c(v, u)$ , the costs to  $u, x$ , and  $w$  will all change, as their paths go through  $u$ . This would trigger an update.

\*  $c(v, w) \geq 4$  does not change the distance vector.

**C.** Yes there will be a count-to-infinity problem and stabilised after 3 message exchanges between  $x$  and  $v$ .

	$D_x(w)$	$D_u(w)$	$D_v(w)$	Explanation
$t = 0$	1	3	4	Initial distances
$t = 1$	10	3	4	$D_x(w) = \min(40, 6 + D_v(w)) = 10$ , $x$ announces $u$ .
$t = 2$	10	12	4	$u$ announces $v$ .
$t = 3$	10	12	13	$v$ announces $x \rightarrow \textcircled{1}$
$t = 4$	19	12	13	$x$ announces $u$ .
$t = 5$	19	21	13	$u$ announces $v$ .
$t = 6$	19	21	20	$D_v(w) = \min(20, 21 + 1)$ , $v$ announces to $x, u \rightarrow \textcircled{2}$
$t = 7$	26	21 (via $x$ )	20	$x$ announces to $u$ .
$t = 8$	26	21 (via $v$ )	20	$D_u(w) = \min(26 + 2, 1 + D_v(w))$ , $u$ announces to $x$ .
$t = 9$	23	21	20	$x$ announces $v \rightarrow \textcircled{3}$ .

**D.**

$$40 \leq D_v(w) + c(x, v)$$

$$40 \leq 4 + c(x, v)$$

$$\implies c(x, v) \geq 36$$

13. [6 points]: Answer the questions below. Write concise answers.

- A. Suppose the network layer were to perform routing based on Ethernet MAC addresses instead of IP addresses. What is the major challenge with this approach?
- B. Message complexity is the total number of control messages exchanged (including forwarded copies) until convergence. What is the message complexity of the link state routing algorithm? Suppose a link cost changes, what is the message complexity of the update?
- C. Suppose there are four routers between a source host and a destination host. Ignoring fragmentation, an IP datagram sent from the source host to the destination host will travel over how many interfaces? How many forwarding tables will be indexed to move the datagram from the source to the destination? Suppose the TTL at the source was  $k$ , what is the TTL at the destination?

**Solution:**

- A. The major challenge is **scalability**. MAC addresses are **flat identifiers** (unique but not hierarchical), so routing tables would need an entry for every single device in the world. IP addresses are **hierarchical** (network + host parts), which allows aggregation into prefixes, making routing feasible across the Internet.

- B. Let  $N$  be the number of routers (vertices) and  $E$  the number of links (edges).

*Message complexity of initial convergence:*

Total origins of LSPs =  $N$

Maximum number of LSPs that pass over 1 link (considering both directions) =  $2N$

Maximum number of LSPs that pass over all  $E$  links =  $2NE$

$\Rightarrow$  Message complexity of initial convergence =  $2NE = O(N \cdot E)$

*Message complexity of update after a link-cost change:*

Total origins of LSPs = 2

Maximum number of LSPs that pass over 1 link (considering both directions) = 4

Maximum number of LSPs that pass over all  $E$  links =  $4E$

$\Rightarrow$  Message complexity of update =  $4E = O(E)$

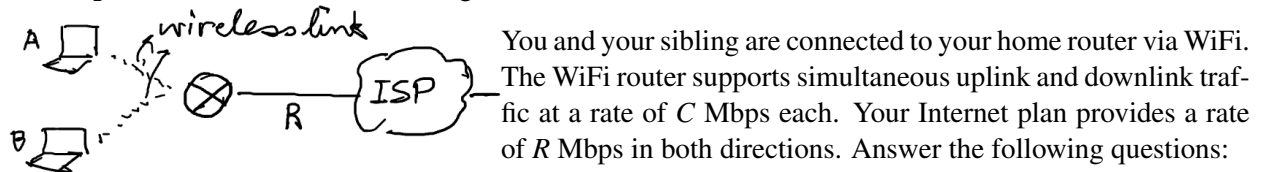
- C. With 4 routers between source and destination, there are 5 links (source  $\rightarrow$  R1, R1  $\rightarrow$  R2, R2  $\rightarrow$  R3, R3  $\rightarrow$  R4, R4  $\rightarrow$  destination). That means the datagram passes through **5 input interfaces + 5 output interfaces = 10 interfaces total**.

With 4 routers, there are exactly 4 lookups in total. Therefore, **4 forwarding tables will be indexed**. 5 is also a possible answer if you consider a lookup at the source host.

Each router decrements TTL by 1 when it forwards a packet. At each one of the 4 routers the TTL is decremented by 1 before forwarding it, so the packet arrives at the **destination host with  $TTL = k - 4$** .

### III Design Questions

14. [8 points]: Consider the following scenario:



- A. You (A) send two packets back-to-back, each of size  $L$ , destined for your sibling's machine (B). Assume the packets are routed via your WiFi router. Calculate the arrival times of the first and second packet at your sibling's machine. Assume propagation, processing, and link-layer delays are negligible, and there is no other traffic.
- B. Suppose you want to verify that the rate you are receiving from your ISP is close to your subscribed rate. Sketch a measurement technique you could use as an end host. You cannot modify your home router and have access only to your machine.
- C. You notice significant lags in your video conferencing call and suspect that your sibling is sending a large amount of data over the Internet. You want to verify this independently. Your router supports priority queuing with high and low priority queues, and by default, data is sent at low priority. Sketch a measurement technique to determine if there is significant cross-traffic on your router.

**Solution:**

**A. Arrival times for 2 back-to-back L size requests:**

- \* First Packet arrives at  $\frac{2L}{C}$
- \* Second Packet arrives at  $\frac{3L}{C}$

**B. Verify ISP rate:**

- \* If  $R > C$  —  $C$  bottleneck — cannot determine ISP rate effectively
- \* Assume  $R < C$ . Send 2 ping packets to ISP (or packets with TTL = 2), back-to-back and measure the time taken to receive the response. Let Packet size =  $L$ .

\*

$$\text{First packet time} = \frac{L}{C} + \frac{L}{R} + \frac{L}{R} + \frac{L}{C} = 2\frac{L}{C} + 2\frac{L}{R}$$

- \* Second Packet has to wait at wireless link until transmission of first packet

\*

$$\text{Second packet time} = \frac{L}{C} + \frac{L}{R} + \frac{L}{R} + \frac{L}{R} + \frac{L}{C} = 2\frac{L}{C} + 3\frac{L}{R}$$

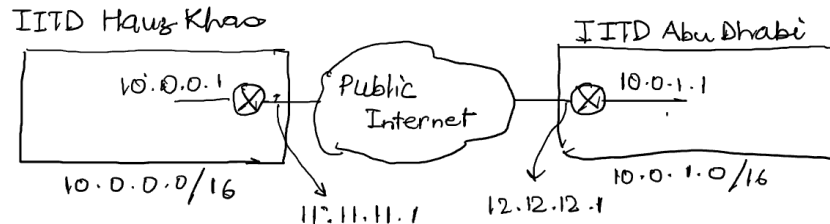
- \* Difference is  $\frac{L}{R}$

- \* Get the difference in time. Knowing packet size  $L$ , you can estimate  $R$

**C. Detecting cross-route traffic:**

- \* i. Send two back-to-back ping packets, one with high priority, and second with low priority. If you observe significant lag/difference, the lower priority buffer/queue is being hogged due to sibling's activity

**15. [4 points]:** Consider the following scenario: The hosts in IITD Hauz Khas campus are in the private network 10.0.0.0/16, while hosts in IITD Abu Dhabi campus are in the private network 10.0.1.0/16. Assume that there is 1 border router in each campus that connects it to the public Internet as shown below. As a network operator, you want to enable seamless communication between hosts in the two campuses *using their private IP addresses*. Design a mechanism that achieves this without requiring any participation from the end hosts.



**Solution: Host-transparent inter-campus connectivity using IP tunnelling:**

*Idea.* Create a point-to-point IP-in-IP tunnel between the two border routers and use simple policy rules to decide which packets go into the tunnel. End hosts keep using their private addresses; only the border routers encapsulate/decapsulate.

**For example, Packet flow (HK → AD)**

- A. Host 10.0.a.b sends an IP packet to 10.1.x.y.
- B. HK border router's policy matches; it encapsulates the entire inner packet [10.0.a.b → 10.1.x.y] inside an outer IP header [11.11.11.1 → 12.12.12.1] and forwards it across the public Internet.
- C. AD border router receives the outer packet, decapsulates (removes the outer header), recovering the original inner packet, and forwards it internally to 10.1.x.y using normal routing.

**Return path** is identical (AD policy matches, encapsulate to 12.12.12.1 → 11.11.11.1, decapsulate at HK, deliver inside 10.0.0.0/16).

**16. [7 points]:** Answer the following questions:

- A. Recall that the data plane is the part of a network device responsible for forwarding data. Compare circuit switching and packet switching in terms of data plane complexity.
- B. Consider the following scenario:
  - i. A backbone network has routers processing packets at 100 Gbps. Assume the average packet size is 500 bytes. What is the maximum budget for different forwarding plane functions?
  - ii. Consider IPv6 routing that performs lookups using unibit tries with an SRAM lookup time of 1 ns. Would it be possible to ensure that the data plane overhead remains within the maximum budget calculated earlier? Explain.
  - iii. Recall that IPv6 includes a 20-bit flow label in its header. How could this header be used to design a faster data plane?

**Solution:****Budget:**

- Data rate = 100 Gbps
- Packet size = 500 Bytes
- 

$$\text{Packets per second} = \frac{10^{11}}{500 \times 8} = 25 \times 10^6$$

·

$$\text{Budget (Time per packet)} = \frac{1}{25 \times 10^6} = 40 \text{ nanoseconds}$$

**SRAM feasibility**

- IPv6 header: 128 bits
- 128 lookups = 128 ns > 40 ns Budget. **Not** feasible in the worst case.

**IPv6 Flow Label**

- A source node adds a 20-bit flow label to packets heading to the same destination.
- This flow label allows routers downstream to identify packets belonging to the same flow based solely on the flow label bits.
- Consequently, routers can forward these packets without performing repeated and expensive lookups, improving forwarding efficiency and reducing processing load.
- **Note:** Routers do not perform a bit-wise lookup of the flow label (i.e., they do not do 20 separate lookups). It is effectively one lookup per packet flow, not 20 lookups (Imagine a Hash Table).