Problem 1

Suppose two packets arrive to two different input ports of a router at exactly the same time. Also suppose there are no other packets anywhere in the router.

(a) Suppose the two packets are to be forwarded to two different output ports. Is it possible to forward the two packets through the switch fabric at the same time when the fabric uses a shared bus?

(b) Suppose the two packets are to be forwarded to two different output ports. Is it possible to forward the two packets through the switch fabric at the same time when the fabric uses switching via memory?

(c) Suppose the two packets are to be forwarded to two different output ports. Is it possible to forward the two packets through the switch fabric at the same time when the fabric uses a crossbar?

Write your solution to Problem 1 in this box

(a) No, it is not possible. This is because only one packet can move through the bus at a given time. Therefore, only one packet can be forwarded through the switch fabric at a time - both packets cannot be forwarded together.

(b) It is not possible. This is because only 1 packet can be read from or written to memory at a given time. So, both packets cannot be forwarded through the switch fabric together.

(c) Yes, since the crossbar is designed to allow multiple packets to be transported simultaneously. It specifically works because the packets arrive at different output ports.

Problem 2

255 · 255 · 255 · 0

No. of bits in the subnet mask → 24 1's

Consider a router that interconnects three subnets: Subnet 1, Subnet 2, and Subnet 3. Suppose all of the interfaces in each of these three subnets are required to have the prefix 224.1.17.0/24. Also suppose that Subnet 1 is required to support at least 60 interfaces, Subnet 2 is to support at least 90 interfaces, and Subnet 3 is to support at least 8 interfaces. Provide three subnet addresses (of the form a.b.c.d/x) that satisfy the constraints. You may use the following link to help verify your result: `http://jodies.de/ipcalc`.

Write your solution to Problem 2 in this box

224·1·17·0

8bits

Valid hosts :- 2^#of host bits - 2

Subnet 1 → 60 interfaces → 6 bits — 2

Subnet 2 → 90 interfaces → 7 bits — 3

Subnet 3 → 8 interfaces → 3 bits — 1

Must be assigned from smallest to largest

Last 8 bits

Subnet 3: 00000000 ← Network Address

: 224·1·17·0/28

11110000 ← Net mask

Subnet 1: 01000000

: 224·1·17·64/26

11000000

Subnet 2: 10000000 : 224·1·17·128/25

10000000

∴ Subnet 1 supports 64 interfaces (62 hosts, 2 reserved),
Subnet 2 supports 128 interfaces (126 hosts, 2 reserved),
Subnet 3 supports 16 interfaces (14 hosts, 2 reserved)

Problem 3

Consider sending a datagram with total length 2400 B into a link that has an MTU (maximum transmission unit) of 800 B. Suppose the original datagram is stamped with the identification number 421 and all IP headers are 20 bytes.

(a) How many fragments are generated?

(b) What are the values in the various fields (header length, total length, identification, MF flag, and fragment offset) in the IP datagram(s) generated related to fragmentation?

Write your solution to Problem 3 in this box

Total length = 2400 B
MTU = 800 B
IP Header = 20 bytes

(a) 800 - 20 = 780 B data per datagram
2400 - 20 = 2380 B overall data payload

No. of fragments = $\left\lceil \frac{2380}{780} \right\rceil \cong \lceil 3.05 \rceil = \boxed{4}$

(b)

Fragment	Header Length	Total Length	ID	MF Flag	Frag. Offset
Fragment 1	20	776+20=796	421	1	0
Fragment 2	20	776+20=796	421	1	97
Fragment 3	20	776+20=796	421	1	194
Fragment 4	20	52+20=72	421	0	291

- 800 - 20 = 780 → closest multiple of 8 bytes is 776 (for total length)
  ↳ Total length can be ≤ MTU
- 776 × 3 = 2328
  2380 - 2328 = 52 bytes (payload size for last segment)

Problem 4

Please answer the following questions regarding checksum.

(a) Why is the IP header checksum recalculated at every router?

(b) What is covered by IP checksum and TCP checksum?

Write your solution to Problem 4 in this box

(a) This is because the IP header changes at each router because the 8-bit Time-to-Live (TTL) field in the header decrements by one at every hop (every router).

(b) IP checksum only checks for possible corruption/errors in the IP header. If the header is corrupted, the packet is dropped, so a bad address should never even be delivered to the host. IP checksum does not examine the payload. TCP checksum checks for possible corruptions in both the TCP header as well as the payload. For the most part, the TCP checksum is only executed by the end destination, while IP checksum is run at each router in the path from host to destination.

Problem 5

In this problem we will explore the impact of NATs on P2P applications. Suppose a peer with username Arnold discovers through querying that a peer with username Bernard has a file it wants to download. Also suppose that Bernard and Arnold are both behind a NAT. Try to devise a technique that will allow Arnold to establish a TCP connection with Bernard without application-specific NAT configuration. If you have difficulty devising such a technique, discuss why.

Write your solution to Problem 5 in this box

This would be quite difficult without an application-specific NAT configuration. This is because there is no mapping between Bernard's LAN IP address and his NAT IP address. Therefore, when Arnold tries to setup a TCP connection with Bernard, he will send a request to Bernard's NAT IP address. More specifically, when Arnold sends a TCP handshake SYN packet to Bernard, the NAT on Bernard's side will fail to determine which host the data is for. This is largely because the NAT table is not yet established for Bernard, especially because there are no outgoing connections.

So, Bernard would need to specify an incoming TCP connection to his NAT. Then, Bernard would need to tell Arnold the NAT port number even before a connection is set up. This would require application-specific NAT coding.