COL334 Midsem

Viraj Agashe

TOTAL POINTS

22.5 / 3

QUESTION 1

1 Q1 9.5 / 1

Q. 1.a)

✓ **+ 1 pts** **1.a Verify Card and pin: req+ response**

✓ **+ 0.5 pts** **1.a Balance query from client: response from server**

✓ **+ 0.5 pts** **1.a Withdrawal normal Case**

✓ **+ 0.5 pts** **1.a Withdrawal low balance case**

✓ **+ 1.5 pts** **Common message format (header, body, query type, response code etc.)**

✓ **+ 1 pts** **Underlying reliable protocol with reason**

- 0.5 pts Not Handling Additional query use case in message format

+ 0 pts Otherwise

Q. 1 b)

✓ **+ 0.5 pts** **If the answer is circuit switching**

✓ **+ 1 pts** **If valid explanation provided ex:(given long sessions with predictable smooth bandwidth and bandwidth reservation possible due to known transmission rate).**

✓ **+ 0.5 pts** **If the answer is no congestion control required**

✓ **+ 1 pts** **If proper reasoning with clarification is provided: ex: (even in worst case where all transmit, no queuing due to sufficient bandwidth)**

+ 0.5 pts Not enough clarification of answer or partially correct reasoning.

+ 0 pts Incorrect for both cases.

Q.1 (c)

+ 3 pts correct approach and correct calculation

✓ **+ 1 pts** **correct approach and incorrect calculation**

+ 0.5 pts partial correct

+ 0 pts otherwise

+ 0.5 **Point adjustment**

query types in common message format

QUESTION 2

2 Q2 5.5 / 1

✓ **+ 1 pts** **2(a) Case 1 [k, k+7]**

✓ **+ 1 pts** **2(a) Case 2 [k-8, k-1]**

+ 1 pts 2(a) Correct Explaination / Partial correct / mod 512

+ 0 pts 2(a) incorrect/Unattempted

Part b

+ 1.5 pts Mentioned that Facebook's authoritative DNS returns CDN hostname, which is the main reason the process differs

+ 1 pts Mentioned that two hostnames are queried: first facebook's, then Akamai's to get to the final IP address

+ 0.5 pts Direct TCP connection is established with the IP address returned by Akamai's DNS to fetch the file

+ 0 pts Incorrect/Unattempted

- 1 pts Idea is correct. Details are missing (like it is the DNS which returns information about Akamai or that Akamai's hostname is again queried).

✓ **+ 1.5 pts** **Partially Correct**

+ 1 pts Partially Correct

✓ **+ 1 pts** **Part c.(i)**

✓ **+ 1 pts** **Part c.(ii)**

+ 1 pts Part c.(iii)

QUESTION 3

3 Q3 7.5 / 1

Q.3 (a)

✓ **+ 1 pts** **Correct Network Layer headers**

+ 0.5 pts Partially Correct Network Layer headers

✓ **+ 1 pts** **Correct Transport Layer headers**

+ **0.5 pts** Partially Correct Transport Layer headers

+ **1 pts** Correct Application Layer headers

✓ **+ 0.5 pts** **Partially Correct Application Layer headers**

+ **0 pts** Incorrect/ Not attempted

Q.3 (b)

✓ **+ 0.25 pts** **Correct Address range for Interface 3**

✓ **+ 0.25 pts** **Correct No of Addresses for Interface 3**

✓ **+ 0.25 pts** **Correct Address range for Interface 2**

✓ **+ 0.25 pts** **Correct No of Addresses for Interface 2**

✓ **+ 0.25 pts** **Correct Address range for Interface 1**

✓ **+ 0.25 pts** **Correct No of Addresses for Interface 1**

✓ **+ 0.25 pts** **Correct Address range for Interface 0**

✓ **+ 0.25 pts** **Correct No of Addresses for Interface 0**

+ **0 pts** Incorrect/ Not Attempted

Q.3 (c)

✓ **+ 1.5 pts** **for correctly implementing the slow start phase**

✓ **+ 0.5 pts** **for correctly implementing the congestion detection phase.**

✓ **+ 1 pts** **for correctly reducing the threshold and reimplementing the slow start phase.**

note: it will start with 2 mss again.

+ **0 pts** Incorrect

Q.3 (d)

+ **2 pts** correct solution

✓ **+ 0 pts** **incorrect solution**

Q1. (b) A circuit switched network is more appropriate for this, as the transmission rate is constant & the connection continues for a long time. In such a scenario, it makes sense to have dedicated circuits between sender & receiver, as if we use packet switching, there may be packet drops, etc. which would slow down the application & would not be able to send at steady rate. ⇒ There is also no wastage of the network resources in retransmission.

No, we would not need congestion control, as if sum of all data rates is less than capacitity of links, we can use the links to send data for all the applications (dedicated circuit).

(c) Delay = Processing time + Queuing delay + Transmission time + Propogation delay.

For a 56 byte packet :

$$\text{Delay} = \underbrace{\frac{56}{2.5\times1000}}_{\text{Transmission time for link 1}} + \underbrace{\frac{5000\times10^3}{2.5\times10^8}}_{\text{Prop. delay link 1}} + \underbrace{\frac{56}{2.5\times1000}}_{\text{Transmission time}} + \underbrace{3\times10^{-3}}_{\text{Pack switch delay}} + \underbrace{\frac{4000\times10^3}{2.5\times10^8}}_{\text{Link 2}} + \underbrace{\frac{56}{2.5\times1000} + 3\times10^{-3} + \frac{1000\times10^3}{2.5\times10^8}}_{\text{for link 3}}$$

Calculate

=

Q1. (a) Protocol

1. TCP connection req by client
2. TCP connection setup b/w client - server
3. Client : USERNAME <user>
4. Server : USERNAME check, send PIN
5. Client : PIN <pin>

If incorrect

Server: Auth fail, try again (Max of 3 times)

Correct

Server: Auth check.

6. Start timeout at server of 1 minute, reset if client sends a message.

Client QUERIES

1. C: QUERY : Balance
   S : {Fetch amt from database}
   AMT : XXXXX - -

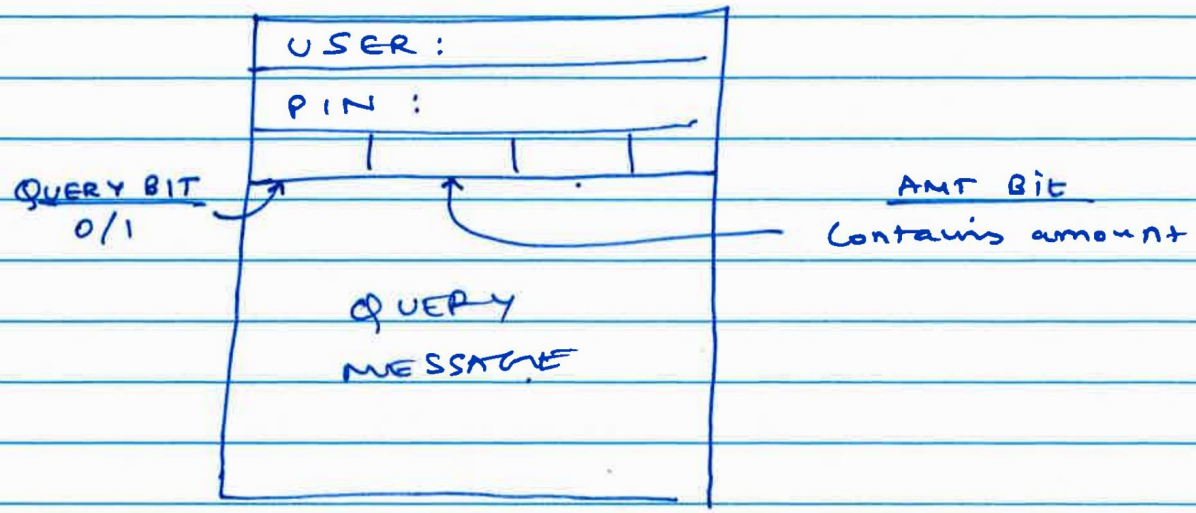
2. C: WITHDRAW AMOUNT YYY - -
   S : {Check from database}

   If YYY - - Y > BALANCE:
   S : ERROR : Withdraw amt. exceed balance

   Else:
   S : Permission for withdraw.
   (To hardware / ATM / etc. if e-currency, send the amount)
   S : Amount <Amt>
   {Update value in database}

→ Underlying transport service would be TCP, as we need reliability as this is sensitive monetary information, packet drop/loss would be undesirable. ~~Data or~~ Data not reaching server/client would be inconvenient for them.

→ Common message:



Generalization → can also use this for depositing amounts.

1 Q1 9.5 / 1

Q. 1.a)

✓ **+ 1 pts** 1.a Verify Card and pin: req+ response

✓ **+ 0.5 pts** 1.a Balance query from client: response from server

✓ **+ 0.5 pts** 1.a Withdrawal normal Case

✓ **+ 0.5 pts** 1.a Withdrawal low balance case

✓ **+ 1.5 pts** Common message format (header, body, query type, response code etc.)

✓ **+ 1 pts** Underlying reliable protocol with reason

- 0.5 pts Not Handling Additional query use case in message format

+ 0 pts Otherwise

Q. 1 b)

✓ **+ 0.5 pts** If the answer is circuit switching

✓ **+ 1 pts** If valid explanation provided ex:(given long sessions with predictable smooth bandwidth and bandwidth reservation possible due to known transmission rate).

✓ **+ 0.5 pts** If the answer is no congestion control required

✓ **+ 1 pts** If proper reasoning with clarification is provided: ex: (even in worst case where all transmit, no queuing due to sufficient bandwidth)

+ 0.5 pts Not enough clarification of answer or partially correct reasoning.

+ 0 pts Incorrect for both cases.

Q.1 (c)

+ 3 pts correct approach and correct calculation

✓ **+ 1 pts** correct approach and incorrect calculation

+ 0.5 pts partial correct

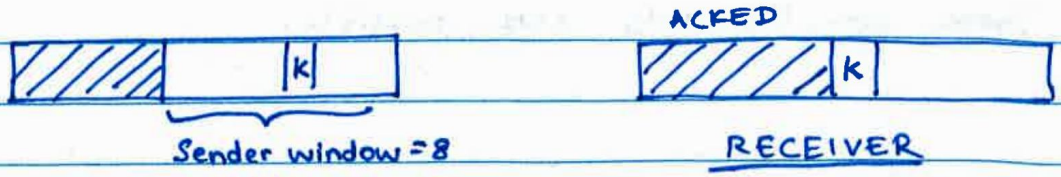
+ 0 pts otherwise

+ 0.5 Point adjustment

query types in common message format

COL334 Minor

Q2. (a)



There can be 8 packets in the senders window. There are 2 possibilities:

① k lies in this window

② k does not lie in the window.

Since receiver has received k-1, either k-1 has not been ACK-ed or it has been acked.

Possibility 1.

Possibility 2.

~~Note that if k-1~~

If k is not in the window, it can be at most 1 outside the window, since k-1 was sent & received by receiver.

So, window can be:

$S_1 = \{k-1, \ldots k-8\}$

Otherwise, if k is inside the window & not ACKed, there are 8 positions for it possible:

$\{k, k-1, \ldots k-7\}$

$\{k+1, k, \ldots k-6\}$

$\vdots$

$\{k+7, \ldots k\}$

These are the only sets possible.

(b) Facebook can indicate the clients to download from CDN in 2 ways:

① HTTP redirect: When FB server gets a request, it can send a redirect message to the client & give the address of the CDN.

② DNS way: Facebook can change the address associated with the domain name to the CDN's address directly (at each DNS server it can put the address of the closest CDN).

(c) (i) SEQ NO = 101
Source Port = 23968
Dest Port = 35092

(ii) ACK = 101
Source Port = 35092
Dest Port = 23968

(iii) ACK = 163

---

2 Q2 5.5 / 1

✓ **+ 1 pts** **2(a) Case 1 [k, k+7]**

✓ **+ 1 pts** **2(a) Case 2 [k-8, k-1]**

+ 1 pts 2(a) Correct Explaination / Partial correct / mod 512

+ 0 pts 2(a) incorrect/Unattempted

Part b

+ 1.5 pts Mentioned that Facebook's authoritative DNS returns CDN hostname, which is the main reason the process differs

+ 1 pts Mentioned that two hostnames are queried: first facebook's, then Akamai's to get to the final IP address

+ 0.5 pts Direct TCP connection is established with the IP address returned by Akamai's DNS to fetch the file

+ 0 pts Incorrect/Unattempted

- 1 pts Idea is correct. Details are missing (like it is the DNS which returns information about Akamai or that Akamai's hostname is again queried).

✓ **+ 1.5 pts** **Partially Correct**

+ 1 pts Partially Correct

✓ **+ 1 pts** **Part c.(i)**

✓ **+ 1 pts** **Part c.(ii)**

+ 1 pts Part c.(iii)

Q3.

NETWORK	TRANSPORT	APPLICATION

Q2. (a) NETWORK.

- Source IP : 200.200.200.2000
- Dest IP : 100.100.100.100
- TTL : 35

TRANSPORT

- Port : 34262
- TCP (HTTP is over TCP) (Protocol)

APPLICATION.

- "HTTP/1.1 200 OK\r\n Content - - - - - - - </html>"
- Protocol : HTTP/1.1
- Content Length : 31
- Status code : 200
- Status message : OK
- Data : <html> - - - - - </html>

Q3. (b) Prefix

Q3. (b) Interface 1 → [01000000000, 01111111111]

Range

No. of addresses : 2^{10}

Interface 2 : [010

Q3. (b) Interface 1. → [011 000000000 – 011 111111111]

9

$N_{ADDR} = 2^9$

Interface 2 → [0100 00000000 – 0100 11111111]

8

$N_{ADDR} = 2^8$

Interface 3 → [0101 00000000 – 0101 11111111]

$N_{ADDR} = 2^8$

Interface 4 : Gets all address starting with 00 or with 1.

∴ [10000000000 – 11111111111]

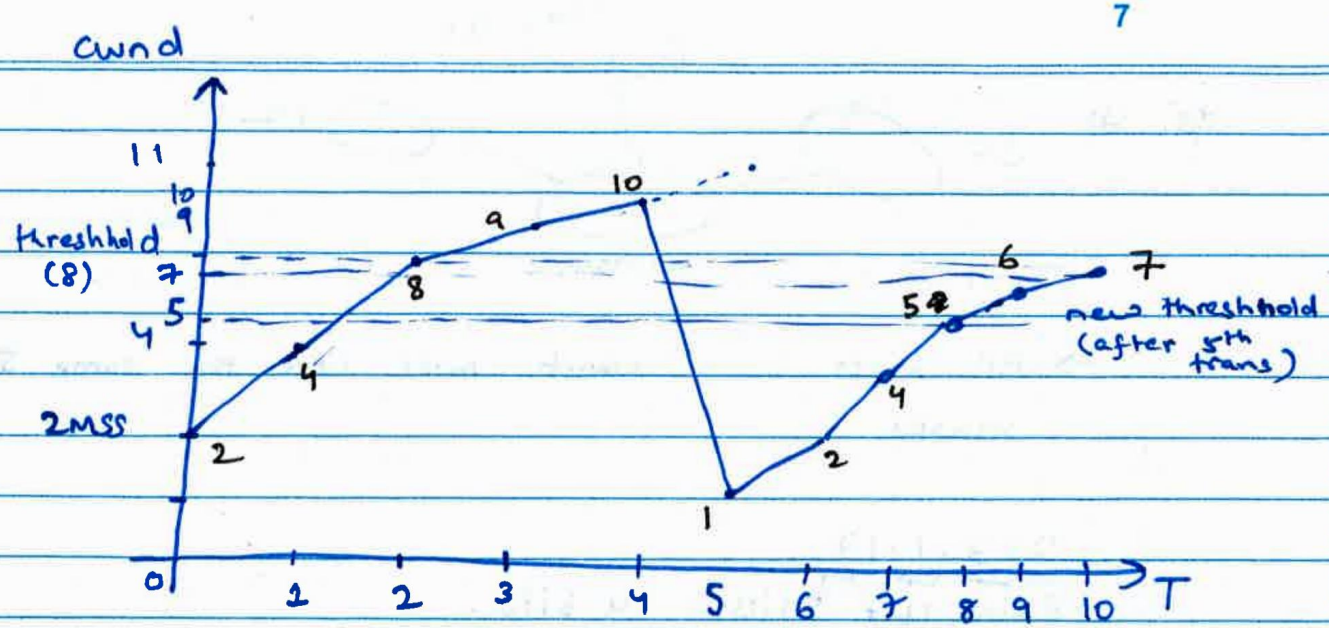
11 11

+ [00 0 – – 0 – 00 1111 . . .]

10

∴ $2^{11} + 2^{10} = 3.2^{10}$ addresses.

Q3. (c)

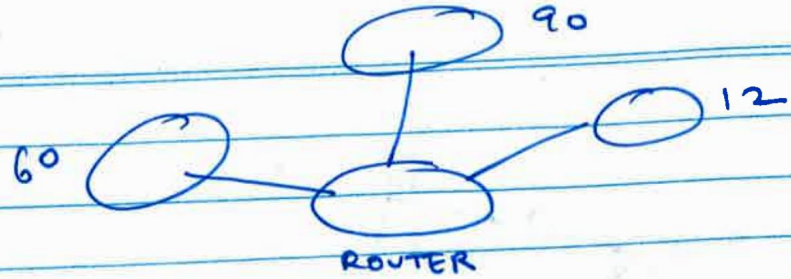


Steps:

- First, till threshhold is reached, cwnd will be doubled.
- When threshhold (8) is reached, now cwnd is added by 1 at each step.
- Packet timeout occurs between T=4 & 5, when cwnd = 10. So, cwnd = 1 MSS, and the threshhold becomes $\frac{cwnd}{2} = 5$.
- Again cwnd is doubled till it reaches threshold. In this case it will be doubled till 4, after 4, 2*4 ≥ 5 so it is set to 5. (after 8th transmission).
- Now assuming no timeout for cwnd = 6 & 7, the cwnd will be incremented by 1. So after 10th transmission,

cwnd = 7

Q3. (d)

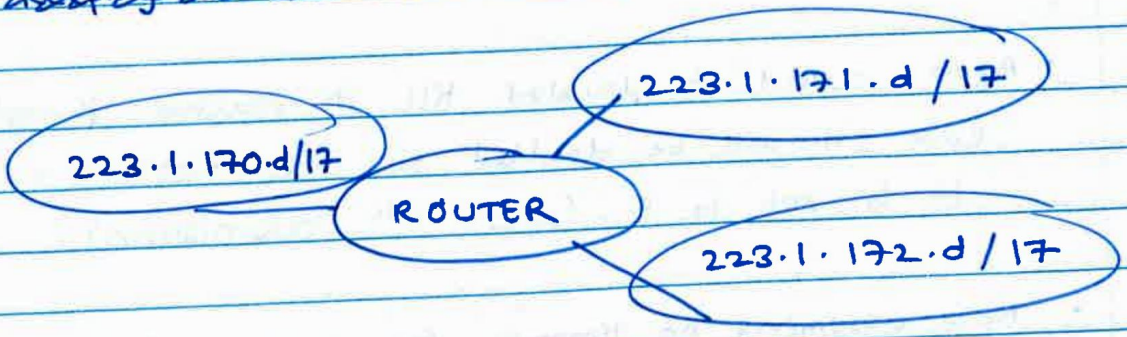


→ All hosts in a subnet must have the same subnet masks

223.1.17
8 bits 1 bit 5 bits = 14 bits.

Host bits needed for 90 addresses ⇒ 7 ($2^7 > 90$)
" 60 addresses ⇒ 6 ($2^6 > 60$)
" 12 ⇒ 4 ($2^4 > 12$)

∴ Addresses :



These are 3 network addresses which satisfy the constraints. (can put any 7 bit no. for d.)

eg.
223.1.170.100
223.171.121
223.172.115

3 Q3 7.5 / 1

Q.3 (a)

✓ + 1 pts Correct Network Layer headers

+ 0.5 pts Partially Correct Network Layer headers

✓ + 1 pts Correct Transport Layer headers

+ 0.5 pts Partially Correct Transport Layer headers

+ 1 pts Correct Application Layer headers

✓ + 0.5 pts Partially Correct Application Layer headers

+ 0 pts Incorrect/ Not attempted

Q.3 (b)

✓ + 0.25 pts Correct Address range for Interface 3

✓ + 0.25 pts Correct No of Addresses for Interface 3

✓ + 0.25 pts Correct Address range for Interface 2

✓ + 0.25 pts Correct No of Addresses for Interface 2

✓ + 0.25 pts Correct Address range for Interface 1

✓ + 0.25 pts Correct No of Addresses for Interface 1

✓ + 0.25 pts Correct Address range for Interface 0

✓ + 0.25 pts Correct No of Addresses for Interface 0

+ 0 pts Incorrect/ Not Attempted

Q.3 (c)

✓ + 1.5 pts for correctly implementing the slow start phase

✓ + 0.5 pts for correctly implementing the congestion detection phase.

✓ + 1 pts for correctly reducing the threshold and reimplementing the slow start phase.

note: it will start with 2 mss again.

+ 0 pts Incorrect

Q.3 (d)

+ 2 pts correct solution

✓ + 0 pts incorrect solution

भारतीय प्रौद्योगिकी संस्थान दिल्ली
INDIAN INSTITUTE OF TECHNOLOGY DELHI

लघु परीक्षा उत्तर पुस्तिका
MINOR TEST ANSWER BOOK

नाम
Name VIRAJ AGASHE

अनुक्रमांक
Entry No. 2020CS10567

पाठ्यक्रम सं.
Course No. COL334

ग्रुप संख्या
Group No.

पाठ्यक्रम शीर्षक
Course Title COMPUTER NETWORKS

लघु परीक्षा सं.
Minor Test No. 1

दिनांक
Date 26-09-2022

प्रयोग किए गए अनुवर्ती पृष्ठों की संख्या
No. of continuation sheets used

प्रश्न सं. Q.No.	प्राप्त अंक Marks
1.	
2.	
3.	
4.	
5.	
6.	
7.	
8.	
9.	
10.	
कुल TOTAL	

पाठ्यक्रम निर्धारक के हस्ताक्षर और दिनांक
Signature of Course Co-ordinator and date

अनुचित साधनों का प्रयोग करने वाले छात्रों को निलम्बित/निष्कासित किया जा सकता है।
Students using unfair means are liable to be punished by Suspension/Expulsion.

परीक्षा केन्द्र में सेलफोन, काम्युनिकेटर्स व पीडीए साधनों का प्रयोग करना सख्त मना है।
Use of cell-phones, Communications & PDAs in the Examination Hall is strictly prohibited.

सभी पृष्ठों पर लिखें। Write on all pages.