भारतीय प्रौद्योगिकी संस्थान खड़गपुर
INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR
Department of Computer Science and Engineering

CS 31006: Computer Networks **Mid Semester Examination**
Full Marks: 50 Time: 2 hours Spring, 2019

Note:

(i) There are THREE questions in this paper. Answer all the questions. The answers should be precise and to-the-point.

(ii) Write down the assumptions clearly, if any. No queries will be entertained during the exam hours.

1. (a) Assume that you are within the IITKGP network, and your name server IP is 202.141.80.81. You want to access a website `www.mynetworkcourse.mooccourses.ac.jp`, which is a newly launched website, and the address information has not been cached in any of the intermediate DNS server. The `ac.jp` name-server maintains the authoritative resource record for this website. Explain with a diagram, how the name resolution will be done when you throw a DNS query for this website name. [4]

Answer: The `ac.jp` name server maintains the authoritative resource record. So, the iterative resolution will first go to 202.141.80.81, then at `.jp` name server and finally at `.ac.jp` name server. The `ac.jp` name server will send back the response with the IP of the website.

(b) What are the purposes of connection establishment at the transport layer? Can the transport layer ensure reliability without establishing a connection? [2+2]

Answer: The connection establishment ensures – (a) both the communication ends are ready for sending/receiving messages; (b) The necessary information for communication synchronization (like the initial sequence number) is exchanged between the communication ends.

Yes, it can – for ensuring reliability, you just need a positive synchronization of sequence numbers between the sender and the receiver (every segment needs to be identified by a unique consecutive monotonic increasing sequence number and the receiver should uniquely identify each segment and acknowledge that); however, the problem is that there needs to be a predefined fixed initial sequence number and every communication should start from that initial sequence number.

(c) Assume that you want to run two different instances of HTTP servers, which will host two different sets of contents. What can be the problem if you run them on the same port address? Assuming that the two servers are hosted on ports 80 and 8080, respectively, and the domain name of your machine is `www.mypages.ac.in`; what will be the two URLs to access the two servers? Consider that you want to access the `index.html` web-page which is hosted in both the servers. [2+2]

Answer: Port number is used for application level service multiplexing. Running two different servers at the same port will create an inconsistency, where the transport layer will not be able to differentiate between the segments originated from or destined to those two servers.

```
http://www.mypages.ac.in:80/index.html
http://www.mypages.ac.in:8080/index.html
```

(d) Consider the following HTML page `mypage.html`.

```
<!DOCTYPE html>
<html>
<head>
    This is a sample page for the CS31006 course
</head>
<body>
    <img src="network.gif" alt="HTML5 Icon" width="128">
    <img src="images/cs31006.gif" alt="HTML5 Icon" width="128">
</body>
</html>
```

How many HTTP GET requests are required to render this page properly on a web browser. Write down those GET requests, assuming that the domain name is `www.mynetworkcourse.org` and you are using HTTP version 1.1. A sample GET request looks like as follows.

```
GET http://www.w3.org/pub/WWW/TheProject.html HTTP/1.1
```

[3]

Answer: Three GET requests as follows.

```
GET http://www,mynetworkcourse.org/mypage.html HTTP/1.1
GET http://www,mynetworkcourse.org/network.gif HTTP/1.1
GET http://www,mynetworkcourse.org/images/cs31006.gif HTTP/1.1
```

2. (a) Why FTP uses a separate control connection to transfer the FTP commands rather than using a single connection for transferring both the commands and the data? [3]

Answer: The data port gets blocked when a large file transfer happens. To send commands even when the data transfer is going on, a separate command/control port is used.

(b) Assume that you initiated a transport layer connection with an initial sequence number 3200 that follows byte sequence number over a 12 bit sequence number space. You have generated 5 data segments, each of size 200 bytes over that connection, and then the connection got crashed. Consider that the lifetime of a segment over the network is 5 seconds, and at every second, you can generate 100 bytes with unique sequence numbers.

After the connection got crashed, say, you have initiated a new connection after 3 seconds from the same port to the same server with an initial sequence number 100. Do you think that this new connection is safe? – Explain with clear justification. [4]

Asnwer: The connection is not safe. 12 bit sequence number can support a maximum of $2^12 = 4096$ distinct sequence numbers that can identify 4096 distinct bytes. With an initial sequence number of 3200, 5 data segments of 200 bytes generate 1000 sequence numbers; so the next starting sequence number becomes at least $4200 - 4095 = 105$. At 3 sec after the crash, the data from the last 200 bytes segment are still in the network, which carries a sequence number till 104. Therefore, a segment with a new sequence number 100 is not safe.

(c) Why can't we use SMTP to retrieve emails from a mail transfer agent (MTA)? I design a modification over SMTP as follows: *The MTA will poll for the user agents (UAs); whenever it will find out that a UA is online, it will use SMTP to push the email to the client mailbox at the UA*. Do you see any problem in this modification of the protocol? [2+2]

Answer: UA needs to come online and then pull the emails from the mailboxes of MTA. SMTP is a push-based protocol.

(1) The MTA needs to poll all the UAs – an overhead for the MTA. (2) The UA will receive the emails only after it has received a poll message and responses to that. This will introduce additional delay and affect service quality.

(d) During the release of a transport layer connection, why a timer with a predefined timeout is started after sending the connection release message? Does this procedure ensure that no in-flight data will be lost during the transmission? [2+2]

Answer: Connection closure is asynchronous – there is no guarantee that the other end will receive the connection closure or the corresponding ACK message. To reduce the probability of delayed in-flight data loss, the timeout is used.

The timeout never guarantees that the complete data will be received, it just reduces the possibility of data loss. Some segments may observe a very high delay and reach after the timeout expires.

3. (a) Consider a sliding window based ARQ with selective acknowledgments to acknowledge out-of-order received segments, where the sender window size is M, and receiver window size is N. The protocol uses a p bit sequence number space. What will be the maximum value of M such

that a sequence number wrap-around does not create problem in distinguishing between a fresh segment and a re-transmitted segment? Give justification for your answer. Consider that both N and p are fixed and cannot be tuned. [5]

Answer: With a p bit sequence number, the maximum sequence number is $2^p - 1$. For selective repeat ARQ, M should be less than $\frac{2^p-1+1}{2} = 2^{p-1}$. Further, $M \leq N$ to ensure no loss from receiver buffer. Therefore, $M = min(N, 2^{p-1})$.

(b) Consider a Go-Back-N based sliding window flow control algorithm (receiver window size is 1) running over a 1 Gbps link. The end-to-end delay between the sender and the receiver is 20ms. What should be the minimum window size in bytes (express in power of 2) to ensure maximum link utilization under this environment? Considering byte sequence numbers, what should be the maximum number of bits in the sequence number space to ensure correct operation of the protocol? [3+3]

Answer: BDP = 2.5 MB, Minimum window size = 2BDP+1 = 5 MB + 1 MB. This value is in between 2^{23} and 2^{22}. So, we can keep the window size as 2^{23}.

To support the above window size, we need at least $2^{23} + 1$ distinct sequence numbers. So, we need 24 bit sequence number.

(c) In sliding window based flow control algorithms, the sender window size is kept no more than the receiver advertised window size. Explain, how this can lead to a potential deadlock situation. How does the algorithm need to be tuned to come out of the deadlock? [2+1]

Answer: Say the sender has received all the ACKs from the receiver and then sends a segment which fill up receiver buffer completely. In that case, the receiver will send the ACK with an available buffer size is zero; which will block the sender. However, until the receiver receives more data packet from the sender, it will not send any further ACK to the sender. So, the receiver is blocked on data, and the sender is blocked on ACK. This will result in a deadlock.

If a window advertisement of zero has been sent to the sender, then the receiver must send another duplicate ACK with available window size, when it has some free space in its buffer.

(d) The general principal of congestion control at the transport layer is to detect congestion once it happens; and then response to it to come out of the congestion. This is a restrictive approach. Explain, why can't we simply avoid the congestion (ensure that congestion does not happen at all) by controlling the sending rate from the transport layer? [3]

Answer: Multiple connections are multiplexed in the intermediate routers; so, it becomes impossible for the transport layer to find out the actual bandwidth a priori, that can eliminate the congestion.

(e) Why do we prefer adjusting the sender window based on an additive increase multiplicative decrease algorithm, for ensuring congestion control in the network? Explain with an example. [3]

Answer: AIMD ensures maximum efficiency with max-min fairness among the contending flows at the steady state. [*Describe with two flows – check the slides*]