CMPEN362 — Practice Midterm Exam Solutions

Name: (all capital letters) ___________________________
Student email: ____
Section: (circle one) Section 1 Section 2

P1	/10
P2	/10
P3	/10
Total	/30

Instructions: Justify your answers. **Answer in the space allotted. Avoid writing too close to the edge of pages.** Do NOT write answers elsewhere (writing not included in scanned exam cannot be graded).

Problem 1

Check all the correct answers. [1 pt per question]

1. From a system view, the Internet consists of a network of interconnected ISP networks and:
   ☐ hosts ☐ packet switches ☐ communication links
   ✓ protocols ☐ programming interfaces ✓ Internet standards

2. In terms of whether the connection between hosts and the access router is shared or dedicated:
   In DSL network, it is ☐ shared ✓ dedicated
   In cellular network, it is ✓ shared ☐ dedicated

3. The two key network-core functions are
   ☐ circuit switching and packet switching ☐ storing and forwarding
   ✓ forwarding and routing ☐ routing and transmission control

4. The four sources of packet delay at a single hop include processing delay, transmission delay, and
   ☐ switching delay ✓ propagation delay ☐ table lookup delay
   ✓ queueing delay ☐ decoding/coding delay

5. The three tiers of Internet core are: Tier-1 ISPs/large content provider networks,
   ☐ national ISPs ✓ regional ISPs
   ☐ home access network ✓ access ISPs

6. T (True) or F (False): Client process is a process running on a client host, and server process is a process running on a server host.
   ☐ T ✓ F

7. T or F: Any single-bit error can be detected by checksum, but a multi-bit error may not.
   ✓ T ☐ F

8. A UDP socket is uniquely identified by
   ☐ source IP address and source port number
   ✓ destination IP address and destination port number
   ☐ source and destination IP addresses, and source and destination port numbers

9. What are the reasons for an application to prefer UDP over TCP?
   ✓ no connection establishment delay
   ✓ no throttling due to congestion control
   ✓ smaller header
   ☐ reliable delivery

10. Fill in blanks: Suppose host A successfully sends a TCP segment to host B with sequence number 22, acknowledgement number 89, and a payload of 120 bytes. The return segment from B to A will have sequence number __89__ and acknowledgement number __142__.

Problem 2

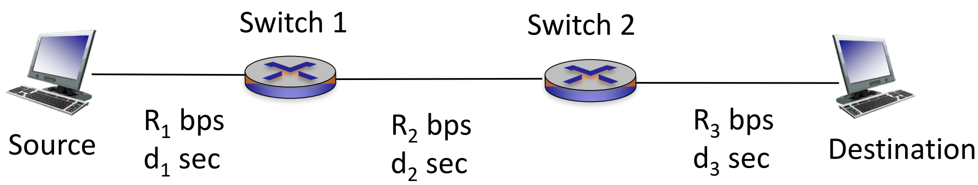

Figure 1: Problem 2.a illustration.

a) Consider a source-destination pair connected by 2 packet switches via 3 links as illustrated in Fig. 1. Suppose link i $(i = 1, 2, 3)$ has a bandwidth of R_i bps and a propagation delay of d_i seconds. Ignore queuing and processing delays.

(i) How long does it take to move an M-bit message from source to destination without message segmentation? [1 pt]

$$\sum_{i=1}^{3} \left(\frac{M}{R_i} + d_i \right)$$

(ii) Suppose that the message is segmented into P packets of equal length and $R_i \equiv R$ $(i = 1, 2, 3)$. How long does it take for the first packet to arrive at the destination? [1 pt] How long does it take for all the packets to arrive at the destination? [1 pt]

First packet delay: $\sum_{i=1}^{3} \left(\frac{M}{PR_i} + d_i \right) = \frac{3M}{PR} + \sum_{i=1}^{3} d_i$

Total delay: $\frac{3M}{PR} + \sum_{i=1}^{3} d_i + (P-1)\frac{M}{PR} = \frac{(P+2)M}{PR} + \sum_{i=1}^{3} d_i$

(iii) Now suppose $R_1 > R_3 > R_2$. Repeat the calculation in b). [2 pt]

First packet delay: $\sum_{i=1}^{3} \left(\frac{M}{PR_i} + d_i \right)$

Total delay: $\sum_{i=1}^{3} \left(\frac{M}{PR_i} + d_i \right) + (P-1)\frac{M}{PR_2}$ (because the bottleneck link capacity is R_2)

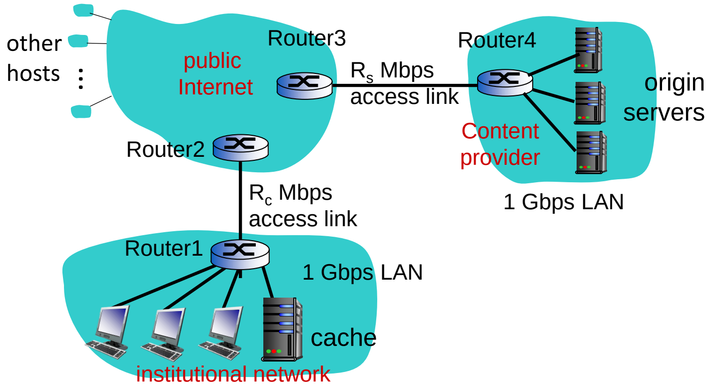

Figure 2: Problem 2.b illustration.

b) Consider hosts in an institutional network accessing web content as in Fig. 2. Suppose each object is 80K bits, hosts in the institutional network generate 18 requests/sec, and hosts from other parts of the Internet generate 100 requests/sec. Suppose that the total delay for Router2 to send a request to Router3 and Router3 to send the response back to Router2 is 2 seconds. Ignore propagation delays for access and LAN links. Let $R_c = 1.54$ Mbps and $R_s = 20$ Mbps. Answer to precision 0.001 second.

(i) Find the total delay for obtaining one object from the origin server to a host in the institutional network. [2 pt]

Total delay = server LAN delay + server access delay + Internet delay + client access delay + client LAN delay.
server/client LAN delay = 0.08 Mbits/1 Gbps = 0.08 ms
server access delay $= \frac{1}{(20Mbps/0.08Mbits/obj)-118obj/s} \approx 7.6$ ms
client access delay $= \frac{1}{(1.54Mbps/0.08Mbits/obj)-18} = 0.8$ s
$\Longrightarrow$ total delay $= 0.08ms + 7.6ms + 2s + 0.8s + 0.08ms \approx 2.808$ s

(ii) Now deploy a web cache in the institutional network with hit rate 0.2. What is the average object downloading delay for hosts in the institutional network? [3 pt]

Hit delay = LAN delay = 0.08 ms
Miss delay = server LAN delay + new server access delay + Internet delay + new client access delay + client LAN delay
new server access delay $= \frac{1}{(20/0.08)-100-18\times 0.8} \approx 7.4$ ms
new client access delay $= \frac{1}{(1.54/0.08)-18\times 0.8} \approx 0.206$ s
miss delay ≈ 2.214 s
$\Longrightarrow$ average delay $= 0.2\cdot$ hit_delay $+ 0.8\cdot$ miss_delay ≈ 1.771 s

Problem 3

Hosts A, B, and C want to send segments to Host S. Each of A, B, and C is connected to S via a channel that can lose/corrupt (but not reorder) segments. Design a stop-and-wait transport protocol to make sure that S's application layer receives segments in the order of: A, B, C, A, B, C...

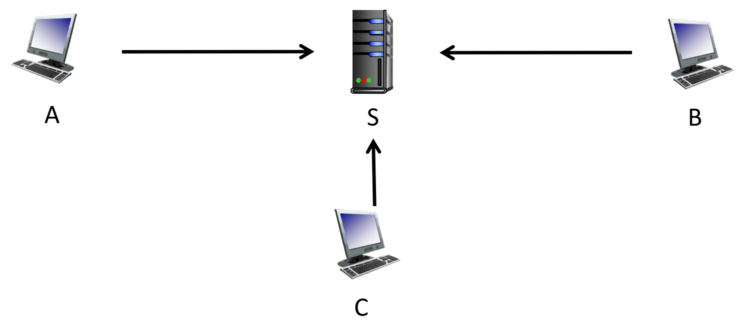

Figure 3: Problem 4 illustration.

You can use the following procedure calls:
– `rdt_send(data)`: called by upper layer to send data in 'data';
– `rdt_rcv(rcvpkt)`: called by lower layer after receiving packet 'rcvpkt';
– `from_host(rcvpkt, hostid)`: true if packet 'rcvpkt' is from host 'hostid';
– `has_seq(rcvpkt, seqnum)`: true if packet 'rcvpkt' has sequence number 'seqnum';
– `corrupt(rcvpkt)`: true if packet 'rcvpkt' is corrupted;
– `udt_send(sndpkt, hostid)`: call lower layer to send packet 'sndpkt' to host 'hostid';
– `extract(rcvpkt, data)`: extract payload of packet 'rcvpkt' into data structure 'data';
– `deliver(data)`: call upper layer to deliver data stored in 'data';
– `make_pkt(seqnum, data)`, `make_pkt(seqnum, ACK)`: return a data or acknowledgement packet with sequence number 'seqnum';
– `start_timer`: start timer;
– `stop_timer`: stop timer;
– `timeout`: called when timer runs out.
In addition, use "!" for negation, "&&" for logical AND, and "||" for logical OR.

a) For the FSM at the receiver S as shown in Fig. 4, give the content of states 4, 5, and 6 following the states given in the first row, and describe the meaning of each state. [2 pt]

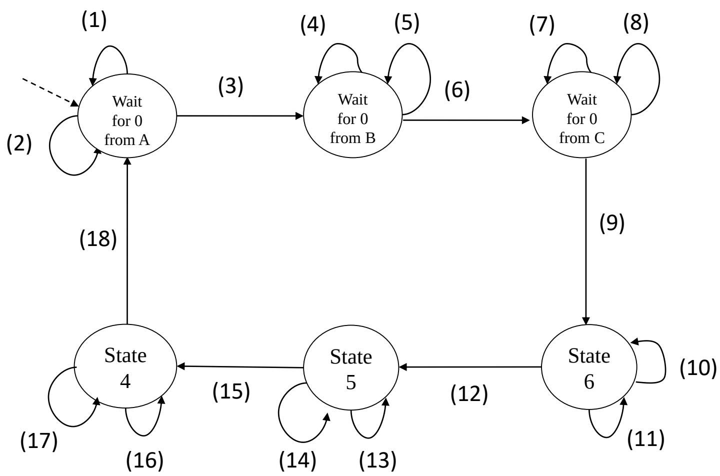

Figure 4: Problem 4: receiver FSM.

State 4: wait for 1 from C (the receiver is waiting for a packet with sequence number 1 from host C)
State 5: wait for 1 from B (the receiver is waiting for a packet with sequence number 1 from host B)
State 6: wait for 1 from A (the receiver is waiting for a packet with sequence number 1 from host A)

b) Complete the event-action list for the following transition links. [4 pt]

(1) `rdt_rcv(rcvpkt) && !from_host(rcvpkt, A)`
Λ

(2) `rdt_rcv(rcvpkt) && from_host(rcvpkt, A) &&`
`(corrupt(rcvpkt) || has_seq(rcvpkt, 1))`
sndpkt = make_pkt(1, ACK)
udt_send(sndpkt, A)

(3) `rdt_rcv(rcvpkt) && from_host(rcvpkt, A) &&`
`!corrupt(rcvpkt) && has_seq(rcvpkt, 0)`
extract(rcvpkt, data)
deliver(data)
sndpkt = make_pkt(0, ACK)
udt_send(sndpkt, A)

c) Can we use the sender FSM of one of the protocols learned in class for A? If so, give the protocol name and its states. [4 pt]

Yes. This is exactly the sender of rdt3.0. It has four states:

(a) wait for call 0 from above

(b) wait for ACK 0

(c) wait for call 1 from above

(d) wait for ACK 1

Note: rdt2.1 does not work because the provided functions do not allow NAK. rdt2.2 is not sufficient because it cannot handle losses. The state names can be different, but the meaning must be “waiting for data from application layer, to be sent with sequence number xxx” and “waiting for acknowledgement from receiver, with acknowledgement number xxx”.