2023 EXAMINATIONS

Part II

COMPUTING AND COMMUNICATIONS – Written Exam [2.5 hours]

SCC.203 Computer Networks

---

*Candidates are asked to answer **THREE** questions from **FOUR**; each question is worth a total of 25 marks.*

Use a separate answer book for each question.

Question 1

1. The tables below are state-machine representations of a simple *transport protocol* that provides 100% reliability over a network layer that may corrupt or lose packets. Read the tables as follows: *In the "CURRENT STATE", if the given "CONDITION" is true, then perform the given "ACTIONS" before moving to state "NEXT STATE".*

	SEND SIDE OF PROTOCOL		
CURRENT STATE	CONDITION	ACTIONS	NEXT STATE
$SEND_0$ (start state)	send(data) // protocol's entry point	sndpkt=make_pkt(0, data) send_to_net(sndpkt); **start_timer**	ACK_0
	recv_from_net(rcvpkt)	NULL // ignore	$SEND_0$
ACK_0	recv_from_net(rcvpkt) && (corrupt(rcvpkt) || isACK(rcvpkt, 1))	NULL // ignore	ACK_0
	timeout	send_to_net(sndpkt) **start_timer**	ACK_0
	recv_from_net(rcvpkt) && not corrupt(rcvpkt) && isACK(rcvpkt, 0)	**stop_timer** //success	$SEND_1$
$SEND_1$	send(data)	sndpkt=make_pkt(1, data) send_to_net(sndpkt); **start_timer**	ACK_1
	recv_from_net(rcvpkt)	NULL // ignore	ACK_1
ACK_1	recv_from_net(rcvpkt) && (corrupt(rcvpkt) || isACK(rcvpkt, 0))	NULL // ignore	ACK_1
	timeout	send_to_net(sndpkt); **start_timer**	ACK_1
	recv_from_net(rcvpkt) && not corrupt(rcvpkt) && isACK(rcvpkt, 1)	**stop_timer** // success	$SEND_0$

	RECEIVE SIDE OF PROTOCOL (initialise sndpkt = make_pkt(1, ACK))		
CURRENT STATE	CONDITION	ACTIONS	NEXT STATE
$RECV_0$ (start state)	recv_from_net(rcvpkt) && not corrupt(rcvpkt) && has_seq(rcvpkt, 0)	deliver(rcvpkt); sndpkt = make_pkt(0, ACK); send_to_net(sndpkt)	$RECV_1$
	recv_from_net(rcvpkt) && (corrupt(rcvpkt) || has_seq(rcvpkt, 1))	send_to_net(sndpkt) // send dup ACK of last pkt correctly received	$RECV_0$
$RECV_1$	recv_from_net(rcvpkt) && not corrupt(rcvpkt) && has_seq(rcvpkt, 0)	deliver(rcvpkt); sndpkt = make_pkt(1, ACK); send_to_net(sndpkt)	$RECV_0$
	recv_from_net(rcvpkt) && (corrupt(rcvpkt) || has_seq(rcvpkt, 1))	send_to_net(sndpkt) // send dup ACK of last pkt correctly received	$RECV_1$

Question 1 continues on next page…

Question 1 continued.

The following primitives are used in the CONDITION and ACTIONS sections:

has_seq(rcvpkt, seqno) TRUE iff *rcvpkt* has sequence number *seqno*
isACK(rcvpkt, seqno) TRUE iff *rcvpkt* is an ACK and has sequence number *seqno*
corrupt(rcvpkt) TRUE iff *rcvpkt* is corrupt
timeout TRUE iff the timer fires

send(data) called by application layer to transmit *data* via our transport protocol
pkt = **make_pkt**(seq, data) make a packet *pkt* with seq number *seq* and payload of *data*
deliver(rcvpkt) extract data payload from *rcvpkt* and pass it up to the application layer
send_to_net(sndpkt) send *sndpkt* to the network
recv_from_net(rcvpkt) pass incoming *rcvpkt* up from the network to our transport protocol
start_timer, stop_timer start/stop the timer

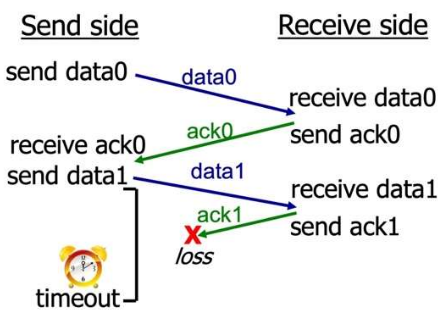
1.a. Describe, perhaps using your knowledge of TCP, how the **corrupt**(rcvpkt) primitive might be implemented.

[5 marks]

1.b. Describe in high level terms how the protocol deals with corrupt packets. Address corruption of both data packets and ACK packets.

[5 marks]

1.c. In the following time sequence diagram (in which time flows from top to bottom) a timeout has just occurred because an ACK has been lost.



Extend the diagram downwards, *being sure to use the same labelling conventions*, to the point that the sender knows for sure that 3 successive data packets (i.e. packets with sequence numbers 0, 1 and 0) have been received (you may assume that no further packet loss occurs). How would your diagram be different if it had been *data1* that had been lost rather than *ack1*?

[5 marks]

Question 1 continues on next page…

Question 1 continued.

1.d. Consider the deployment of our transport protocol in a long-distance (e.g. cross-continental) gigabit network environment. What throughput could it achieve in such an environment assuming a data packet size of 1,000 bits and a round-trip time (RTT) of 250 milliseconds? What is it about the design of the protocol that leads to this level of performance?

[5marks]

1.e. Based on your knowledge of TCP, discuss in outline how our protocol might be upgraded to significantly improve on the result from part (d).

[5 marks]

[Total 25 marks]

Question 2

2.a. Define *transmission delay* and *propagation delay* in terms of link speed, link length and packet size. Make clear the distinction between the two concepts.

[5 marks]

2.b. Discuss *forwarding* and *routing* in the Internet, making clear the distinction between the two concepts.

[5 marks]

2.c. Briefly explain the concept of class-based routing in the Internet (i.e. address classes A, B and C); then discuss Classless Interdomain Routing (CIDR), commenting on why class-based routing is being superceded by CIDR.

[5 marks]

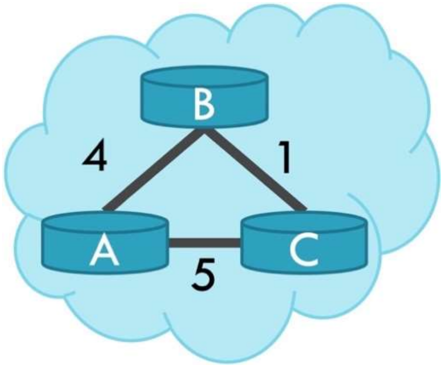
2.d. If a datagram with destination address `192.15.4.65` arrives at a router, which egress port should the router choose if its forwarding table contains the following apparently-relevant entries:

```
192.15.4.0/111111111111111111111111 -> egress port 5
192.15.0.0/1111111111111111         -> egress port 7
```

Demonstrate your understanding of forwarding in CIDR by explaining clearly *why* the router should make this choice.

[3 marks]

2.e. The diagram below shows the initial state of a network for which we want to derive per-node forwarding tables using the Bellman-Ford algorithm.



We can represent initial per-node distance vectors (DVs) in the following manner (this is for node A): <NODE A: Dest B, Cost 4, Next B; Dest C, Cost 5, Next C>

Question 2 continues on next page…

Question 2 continued.

i. Write out initial DVs for nodes B and C in the same way.

[2 marks]

ii. Now suppose that B notices a change in the cost of the link between itself and A: this was initially 4, but has now changed to 1. In response, B updates its DV to:

<NODE B: Dest A, Cost 1, Next A; Dest C, Cost 1, Next C>

Node B then sends its updated DV to A and C. Write out new DVs for A and C that incorporate the new information provided by B.

[3 marks]

iii. Have we reached steady-state by this point? If so, how do we know? If not, explain any further stages that you can identify.

[2 marks]

[Total 25 marks]

Question 3

3.a.

Consider the HTTP request by a web browser (on the left) and the corresponding response (on the right) from the web server captured by Wireshark:

GET /cs453/index.html HTTP/1.1<cr><lf> Host: gai a.cs.umass.edu<cr><lf> User-Agent: Mozilla/5.0 (Windows;U; Windows NT 5.1; en-US; rv:1.7.2) Gecko/20040804 Netscape/7.2 (ax) <cr><lf> Accept:ex t/xml, application/xml, application/ xhtml+xml, text /html;q=0.9, text/plain;q=0.8, image/png,*/*;q=0.5 <cr><lf> *Accept-Language: en-us,en;q=0.5<cr><lf>* *Accept- Encoding: zip,deflate<cr><lf>* *Accept-Charset: ISO -8859-1,utf-* *8;q=0.7,;q=0.7<cr><lf>* *Keep-Alive: 300<cr> <lf>* *Connection:keep-alive<cr><lf><cr><lf>*	HTTP/1.1 200 OK<cr><lf> *Date*: Tue, 07 Mar 2023 12:39:45GMT<cr><lf> *Server:* Apache/2.0.52 (Fedora) <cr><lf> *Last Modified*: Sat, 10 Dec2022 18:27:46 GMT <cr><lf> ETag: ”526c3-f22-a88a4c80”<cr><lf> *Accept- Ranges:* bytes<cr><lf> *Content-Length*: 3874<cr><lf> *Keep-Alive*: timeout=max=100<cr><lf> *Connection*: Keep-Alive<cr><lf> *Content-Type:* text/html; charset= ISO-8859- 1<cr><lf> <cr><lf> <!doctype html **public** ”- *//w3c//dtd html 4.0tr* *ansitional//en”><lf><html><lf>* <head><lf> <m eta http-equiv=”Content- Type” content=”text/html; charset=iso-8859-1”><lf> <meta name= ”GENERATOR” content=”Mozilla/4.79 [en] (Wi ndows NT 5.0; U) Netscape]”><lf> <title>CMPS CI 453 / 591 / NTU-ST550A Spring 2015 homepage

i. What is the URL of the document requested by the browser? Explain how you composed the URL from the request.

[2 marks]

ii. Explain how the client can determine the IP address of the server hosting this document?

[1 mark]

iii. Was the server able to successfully find the document? (Yes/No)

[1 mark]

iv. Explain a possible use of the “Last Modified” field for clients that cache retrieved objects.

[2 marks]

v. What is the size of the document in bytes returned in the response?

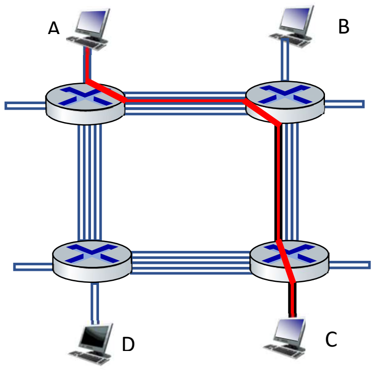
[1 mark]

Question 3 continues on next page…

Question 3 continued.

3.b.

In the diagram below, there are four switches connected through links, each with a capacity of 40 Gbps. Hosts A, B, C, and D are each attached to a switch. A link connecting two switches supports exactly four circuits each with a capacity of 10 Gbps. As an example, a 10 Gbps connection from host A to host C gets the 2nd circuit in the top link and the 1st circuit in the right link as shown with the red line. Answer the following questions based on this diagram.



i. What is the maximum number of simultaneous 10Gbps connections between the four hosts that can be in progress at any one time in this network using circuit switching? Indicate each connection that you include in the count.

[2 marks]

ii. Given that all the 10Gbps connections are between hosts A and C, what is the maximum number of simultaneous connections that can be in progress? Indicate each connection that you include in the count.

[2 marks]

iii. Suppose we want to make four connections between hosts A and C, and another four connections between hosts B and D. Can we route these calls through the four links to accommodate all eight connections simultaneously? Explain how this is possible or why this is not possible.

[2 marks]

iv. How many 5Gbps connections between the hosts attached to A and B can be supported simultaneously with circuit switching and packet switching?

[2 marks]

Question 3 continues on next page…

Question 3 continued.

3.c.

A client needs to perform a transaction with a server that involves the client downloading just a few bytes of data from the server. If this transaction needs to be completed as fast as possible, should the client and server use UDP or TCP as the transport protocol? Explain why.

[3 marks]

3.d.

Consider the scenario where you click on a link to obtain a Web page from your web browser. The IP address for the associated URL is not cached in your local host, so a DNS lookup is necessary to obtain the IP address.

Suppose that 3 DNS servers are visited before your host receives the IP address from DNS; the successive visits incur an RTT of 10ms, 25ms, and 35ms. Further suppose that the Web page associated with the link contains exactly one object, consisting of a small amount of HTML text. Let the RTT between the local host and the server containing the object be 30ms. Assuming zero transmission time of the object, how much time elapses from when the client clicks on the link until the client receives the object? How much time will elapse when the client subsequently clicks on the same link assuming that the IP address for the URL is cached from the first visit? (Hint: With TCP, two RTTs are needed to download a small web page)

[2 marks]

3.e. Consider a portion of a traceroute output below. Indicate whether the output is from the paris-traceroute variant or from the original traceroute tool. Briefly explain how you identified which tool it is.

[3 marks]

Question 3 continues on next page…

Question 3 continued.

```
traceroute to microsoft.com (20.81.111.85), 64 hops max, 52 byte packets
 1  10.32.96.1 (10.32.96.1)  11.901 ms  3.284 ms  3.447 ms
 2  staff-unmanaged.pim.iscore01.rtr.lancs.ac.uk (148.88.254.92)  9.182 ms  3.867 ms  3.867 ms
 3  is-core01.staff-unmanaged.bfw01.rtr.lancs.ac.uk (148.88.250.161)  6.299 ms  4.980 ms  3.624 ms
 4  bfw01.is-border01.rtr.lancs.ac.uk (148.88.253.201)  26.352 ms  17.788 ms  21.526 ms
 5  ae12.manckh-ban1.ja.net (146.97.40.177)  10.310 ms  8.383 ms  5.880 ms
 6  ae11.manckh-sbr2.ja.net (146.97.35.49)  11.479 ms  5.837 ms  6.122 ms
 7  ae0.manckh-ban2.ja.net (146.97.35.190)  11.671 ms  6.702 ms  5.892 ms
 8  ae60-0.man30-96cbe-1b.ntwk.msn.net (104.44.13.30)  10.931 ms  5.861 ms  5.732 ms
 9  104.44.53.151 (104.44.53.151)  15.224 ms
    ae23-0.icr02.lon22.ntwk.msn.net (104.44.239.113)  18.171 ms
    104.44.53.151 (104.44.53.151)  18.692 ms
10  be-122-0.ibr02.lon22.ntwk.msn.net (104.44.21.97)  92.799 ms  88.077 ms
    be-102-0.ibr01.lon22.ntwk.msn.net (104.44.21.89)  91.088 ms
11  be-10-0.ibr01.nyc30.ntwk.msn.net (104.44.18.152)  89.120 ms  86.706 ms  99.867 ms
12  be-9-0.ibr04.bl20.ntwk.msn.net (104.44.28.55)  87.694 ms * *
13  ae160-0.icr01.bl20.ntwk.msn.net (104.44.22.210)  88.484 ms
    ae142-0.icr02.bl20.ntwk.msn.net (104.44.21.224)  88.830 ms
    ae160-0.icr01.bl20.ntwk.msn.net (104.44.22.210)  84.995 ms
14  * * *
15  * * *
```

3.f.

What field in the IP header can be used to ensure that a packet is forwarded through no more than N routers? Explain briefly how this field is used by the traceroute.

[2 marks]

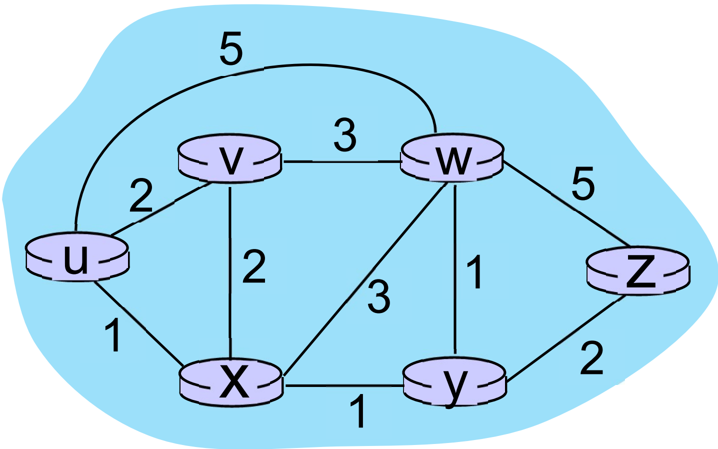
[Total 25 marks]

Question 4

4.a.

Consider the graph model of a network containing the routers u,v,w,x,y, and z with the link costs indicated on each link. With the indicated link costs, use Dijkstra's shortest-path algorithm to compute the shortest path from u to all network nodes. Show how the algorithm progresses and computes the least-cost paths by filling up the table below - the first row is already completed for you. Reminder: N' is the set of nodes already visited (with known least-cost paths), D(k) is the current estimate of the cost of the least-cost path from u to k and p(k) is the predecessor of k on the path from u to k.

[10 marks]



Iteration	*N'*	*D(v),p(v)*	*D(x),p(x)*	*D(w),p(w)*	*D(y),p(w)*	*D(z),p(z)*
1	*u*	*2,u*	*1,u*	*5,u*	∞	∞
2						
3						
4						
5						
6						

Question 4 continues on next page…

Question 4 continued.

4.b.

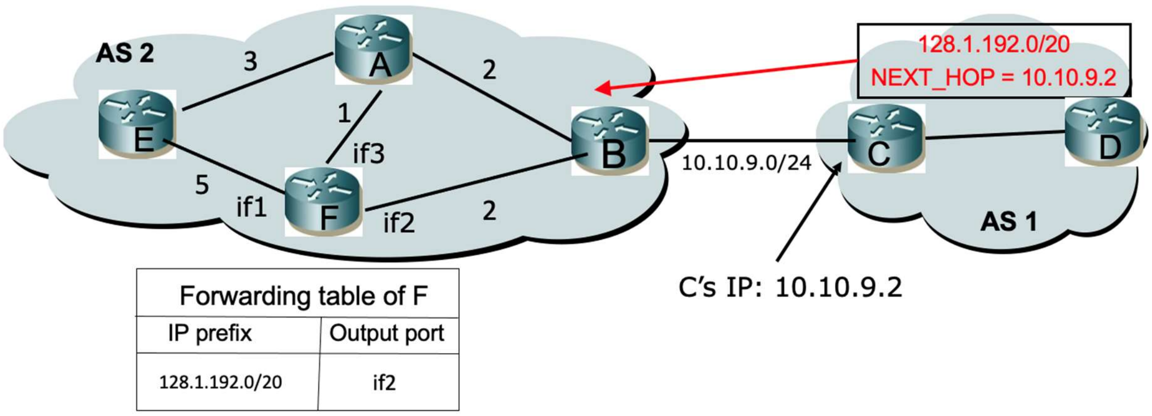
Compute the forwarding table of node u (containing node to next-hop mappings) assuming u uses least-cost paths. Explain how you compute them with a simple algorithm using the predecessor of each node.

[9 marks]

4.c.

Consider the two Autonomous systems depicted below. As shown in the Figure, AS 1 sends AS 2 a BGP advertisement for the prefix 128.1.192.0/20 containing the NEXT_HOP field set to 10.10.9.2. This BGP update then gets propagated inside AS 2. AS 2 uses an intra-domain routing protocol that computes the forwarding tables of routers based on the least-cost paths to destination prefixes (link costs of the links are shown in the Figure). Explain how router F can compute an entry for the prefix 128.1.192.0/20 in its forwarding table?

[6 marks]



[Total 25 Marks]

---End of Paper---