

Problem 1 [10 Pts]

Peterson and Davie, Chapter 3, Exercise 13.

Solution 1

B2's link to bus A is deactivated.

B5 gets deactivated (both of its links are not selected).

B6's link to bus I is deactivated.

Problem 2 [10 Pts]

Peterson and Davie, Chapter 3, Exercise 15.

Solution 2

When A sends to C, all the bridges learn where A is because the frame is broadcasted.

A sends to C:

B1: (A, port A)

B2: (A, port B1)

B3: (A, port B2)

B4: (A, port B2)

When C sends to A, it is routed directly to A, and thus B4 does not learn where C is.

C sends to A:

B1: (A, port A), (C, port B2)

B2: (A, port B1), (C, port B3)

B3: (A, port B2), (C, port C)

B4: (A, port B2)

When D sends to C, B4 broadcasts the packet, and B2 forwards it toward C. B1 thus never learns where D is.

D sends to C:

B1: (A, port A), (C, port B2)

B2: (A, port B1), (C, port B3), (D, port B4)

B3: (A, port B2), (C, port C), (D, port B2)

B4: (A, port B2), (D, port D)

Problem 3 [10 Pts]

Peterson and Davie, Chapter 3, Exercise 19.

Solution 3

(a) Both B1 and B2 add (M, bottom) to their tables and, not knowing where L is, broadcast the packets out of their top interfaces. B1 and B2 then receive each other's packets and forward them out their bottom interfaces. These two packets loop forever.

(b) Initially we (potentially) have four packets: one from M clockwise, one from M counterclockwise, and a similar pair from L. Suppose a packet from L arrives at an interface to a bridge B_i , followed immediately via the same interface by a packet from M. As the first packet arrives, the bridge adds (L, arrival-interface) to the

table (or, more likely, updates an existing entry for L). When the second packet arrives, addressed to L, the bridge then decides not to forward it, because it arrived from the interface recorded in the table as pointing towards the destination, and so it dies. Because of this, we expect that in the long run only one of the pair of packets traveling in the same direction will survive. We may end up with two from M, two from L, or one from M and one from L. A specific scenario for the latter is as follows, where the bridges' interfaces are denoted 'top' and 'bottom':

1. L sends to B1 and B2; both place (L, top) in their table. B1 already has the packet from M in the queue for the top interface; B2 this packet in the queue for the bottom.
2. B1 sends the packet from M to B2 via the top interface. Since the destination is L and (L, top) is in B2's table, it is dropped.
3. B2 sends the packet from M to B1 via the bottom interface, so B1 updates its table entry for M to (M, bottom)
4. B2 sends the packet from L to B1 via the bottom interface, causing it to be dropped. The packet from M now circulates counterclockwise, while the packet from L circulates clockwise.

This solution for (b) taken from Peterson and Davie. Other scenarios resulting in appropriate loops were also accepted.

Problem 4 [20 Pts]

- (a) Peterson and Davie, Chapter 4, Exercise 11.
- (b) Peterson and Davie, Chapter 4, Exercise 14.

Solution 4

(a)

If the timeout is too small, we would fill the network with too many ARP packets trying to map IP addresses to MAC addresses, which don't change too frequently. If the timeout is too long, however, packets may not reach a host if it changes its MAC address (in the case of locally assigned MAC addresses, or if the user replaces his network interface card), if the IP gets assigned to a different computer, etc.

(b)

- a) An ARP query would get broadcasted for every incoming packet, whereas only the first one is necessary.
- b) Keep a list of outstanding ARP queries and do not transmit a new one if there is an outstanding one on the list. We may also retransmit queries on the list after a timeout period if we do not get a reply.
- c) This would cause the first few packets of most new connections to be dropped, requiring that they be retransmitted.

Problem 5 [10 Pts]

Peterson and Davie, Chapter 4, Exercise 21.

Solution 5

- (a) next hop is Interface 0
- (b) next hop is R2
- (c) next hop is R4 (default)
- (d) next hop is R3
- (e) next hop is R4 (default)

Problem 6 [20 Pts]

- (a) Peterson and Davie, Chapter 4, Exercise 34.
- (b) Peterson and Davie, Chapter 4, Exercise 44.

Solution 6

(a) There are many acceptable answers for this question. Here are some of them.

- a) For one, if a router received a packet with an unknown destination, it would resort to broadcasting it. If a new host connected to the Internet and another host tried to communicate to it, the packets would have to be broadcast

across the entire Internet. This cannot scale. Furthermore, routing algorithms such as distance-vector avoid loops by routing along the shortest path. Bridge routing has no notion of shortest path, thus in order to prevent loops we would need to run a spanning-tree-like algorithm on the whole Internet. Even if this did scale, we may have to select a 'root' node and forward all packets toward it.

b) This could scale on a small internal network. In the worst case, it would degrade to flat addressing and behave just like an Ethernet network, except with IP addresses instead of MAC addresses. It would thus be able to scale on any type of network that Ethernet can scale on.

(b)

a) Both DHCP servers would respond to a node's DHCPDISCOVER packet, no matter what subnet it originated from, and assign an IP address from its own subnet. Thus, the subnets that the computers on the LAN would fall in would be determined somewhat arbitrarily, by whichever DHCP response each computer receives first. In practice, DHCP servers are manually configured to ignore requests from MAC addresses of computers on the other subnet.

b) ARP will not be affected. Hosts will only broadcast ARP queries for other hosts on the same subnet; hosts on the other subnet will hear these but won't answer. A host on one subnet would answer an ARP query from the other subnet, if it were ever issued, but it wouldn't be.

Problem 7 [20 Pts]

(a) Run the Bellman-Ford algorithm on the following network to determine the routing table for Router A. Be sure to show A's distance vector at each step.

(b) Show the routing table for Router A at the first step in which its distance vector has no ∞ 's.

(c) Show the routing table for Router A at the final step.

Solution 7

(a)

step 1: (0, 9, ∞ , ∞ , 1, ∞)

step 2: (0, 8, 6, 10, 1, 2)

step 3: (0, 7, 6, 3, 1, 2)

step 4: (0, 4, 6, 3, 1, 2)

step 5: (0, 4, 5, 3, 1, 2)

	<i>Node</i>	<i>Cost</i>	<i>NextHop</i>
	<i>B</i>	8	<i>E</i>
	<i>C</i>	6	<i>E</i>
(b)	<i>D</i>	10	<i>B</i>
	<i>E</i>	1	<i>E</i>
	<i>F</i>	2	<i>E</i>

	<i>Node</i>	<i>Cost</i>	<i>NextHop</i>
	<i>B</i>	4	<i>E</i>
	<i>C</i>	5	<i>E</i>
(c)	<i>D</i>	3	<i>E</i>
	<i>E</i>	1	<i>E</i>
	<i>F</i>	2	<i>E</i>