

## Solutions to Homework#2

1. The following table is cumulative; at each part the VCI tables consist of the entries at that part and also all previous entries.

Exercise part	Switch	Input		Output	
		Port	VCI	Port	VCI
(a)	1	2	0	1	0
	2	3	0	0	0
	3	0	0	3	0
(b)	1	3	0	1	1
	2	3	1	1	0
	4	3	0	1	0
(c)	2	2	0	0	1
	3	0	1	2	0
(d)	1	0	0	1	2
	2	3	2	0	2
	3	0	2	3	1
(e)	2	1	1	0	3
	3	0	3	1	0
	4	2	0	3	1
(f)	1	1	3	2	1
	2	1	2	3	3
	4	0	0	3	2

15. When A sends to C, all bridges see the packet and learn where A is. However, when C then sends to A, the packet is routed directly to A and B4 does not learn where C is. Similarly, when D sends to C, the packet is routed by B2 towards B1 only, and B1 does not learn where D is.

B1: A-interface: A                      B2-interface: C (not D)  
 B2: B1-interface: A                      B3-interface: C                      B4-interface: D  
 B3: B2-interface: A,D                      C-interface: C  
 B4: B2-interface: A (not C)                      D-interface: D

17. (a) When X sends to Z the packet is forwarded on all links; all bridges learn where X is. Y's network interface would see this packet.
- (b) When Z sends to X, all bridges already know where X is, so each bridge forwards the packet only on the link towards X, that is,  $B3 \rightarrow B2 \rightarrow B1 \rightarrow X$ . Since the packet traverses all bridges, all bridges learn where Z is. Y's network interface would not see the packet as B2 would only forward it on the B1 link.
- (c) When Y sends to X, B2 would forward the packet to B1, which in turn forwards it to X. Bridges B2 and B1 thus learn where Y is. B3 and Z never see the packet.
- (d) When Z sends to Y, B3 does not know where Y is, and so retransmits on all links; W's network interface would thus see the packet. When the packet arrives at B2, though, it is retransmitted only to Y (and not to B1) as B2 does know where Y is from step (c). All bridges already knew where Z was, from step (b).
21. (a) If the bridge forwards all spanning-tree messages, then the remaining bridges would see networks D,E,F,G,H as a single network. The tree produced would have B2 as root, and would disable the following links:
- from B5 to A (the D side of B5 has a direct connection to B2)
  - from B7 to B
  - from B6 to either side
- (b) If B1 simply drops the messages, then as far as the spanning-tree algorithm is concerned the five networks D-H have no direct connection, and in fact the entire extended LAN is partitioned into two disjoint pieces A-F and G-H. Neither piece has any redundancy alone, so the separate spanning trees that would be created would leave all links active. Since bridge B1 still presumably *is* forwarding other messages, all the original loops would still exist.
24. A 53-byte ATM cell has 5 bytes of headers, for an overhead of about 9.4% for ATM headers alone.

When a 512-byte packet is sent via AAL3/4, we first encapsulate it in a 520-byte CS-PDU. This is then segmented into eleven 44-byte pieces and one trailing 36-byte piece. These in turn are encapsulated into twelve ATM cells, each of which having 9 bytes of ATM+AAL3/4 headers. This comes to  $9 \times 12 = 108$  bytes of header overhead, plus the 8 bytes added to the CS-PDU, plus  $44 - 36 = 8$  bytes of padding for the last cell. The total overhead is 124 bytes,

which we could also have arrived at as  $12 \times 53 - 512$ ; as a percentage this is  $124/(512+124) = 19.5\%$ .

When the packet is sent via AAL5, we first form the CS-PDU by appending 8 AAL5 trailer bytes, preceded by another 8 bytes of padding. We then segment into *eleven* cells, for a total overhead of  $8 + 8 + 11 \times 5 = 71$  bytes, or  $71/(512+71) = 12.1\%$ .