Rajat Sethi – ECE 4380 – HW5

1.)

S1		S2		S3		S4	
Def.	3	Def.	2	Def.	1	Def.	1
A	1	A	1	C	2	D	2
B	2	B	1	D	3		
		C	3				
		D	3				

2.)

A → B

A → D

B → E

3.)

Append ports (2, 3, 0) to the front of Host A's original list. This will create a port list (2, 3, 0, 3, 0, 1). That way, when the packet reaches Host B, the pathway to Host A is immediately ready. (I assume the ports are read right-to-left, like in the textbook).

4a.)

S2		
Host	**Port**	**Cost**
Def.	0	10
A	0	1
B	1	0
C	0	10
D	0	7

4b.)

New Virtual Circuit Routing Tables:

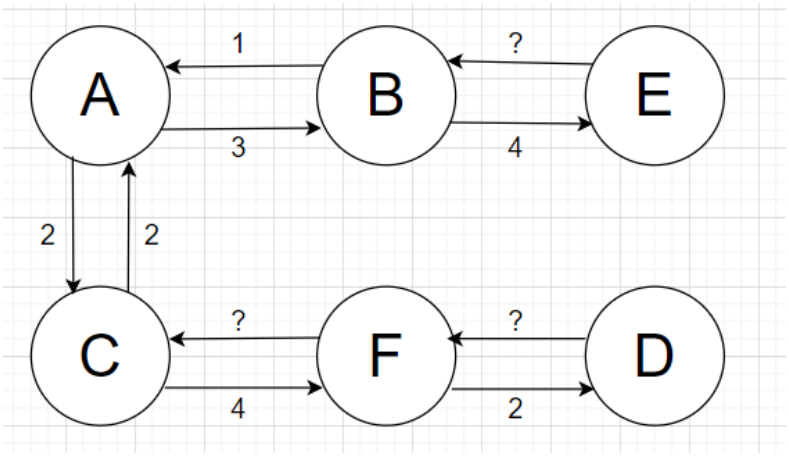
S1				S3			
$Port_{in}$	VCI_{in}	$Port_{out}$	VCI_{out}	$Port_{in}$	VCI_{in}	$Port_{out}$	VCI_{out}
0	0	1	0	0	0	3	0
1	0	0	0	0	1	3	0
2	0	1	1	3	0	0	0

S2					S4			
$Port_{in}$	VCI_{in}	$Port_{out}$	VCI_{out}		$Port_{in}$	VCI_{in}	$Port_{out}$	VCI_{out}
0	0	2	0		0	0	2	0
0	1	1	1		0	1	2	1
2	0	3	0		0	2	1	0
2	1	1	2		2	0	0	0
3	0	1	0					

4c.)

(0, 2, 3, 3)

5a.)



5b.)

The link from C → B = 1

The link from F → D has increased to a cost >7.

The link from E → D = 6

5c.)

A lot of the new links create new paths that are lower cost than the originals. That means almost all of the paths have to be re-updated to find the shortest path. And since path-finding algorithms are quite time-complex, re-determining them all could take a long time for a large network.

6a.)

LAN	LAN 1	LAN 2	LAN 3
1	R	1/2R	1/3R
2	2/3R	R	2/3R
3	1/3R	1/2R	R

LAN 1: NR + NR/2 + NR/3 = 11NR/6

LAN 2: 2NR/3 + NR + 2NR/3 = 7NR/3

LAN 3: NR/3 + NR/2 + NR = 11NR/6

7 * N * (100 Kbps) / 3 = 8 Mbps.

N = 34 Hosts

6b.)

LAN	LAN 1	LAN 2	LAN 3
1	R	3/8R	3/8R
2	3/8R	R	3/8R
3	3/8R	3/8R	R
Backbone	3/4R	3/4R	3/4R

14N * (100 Kbps) /8 = 8 Mbps

N = 45 Hosts

6c.)

Same answer as 6b.) = **45 Hosts**