Problem 1 Solution

a) Use the upper-layer protocol field in IP data gram.

b) 248 = $(11111000)_2$. So we have 2^3 - 2 = 6 IPs can assign, since 11111111 and 11111000 must be reserved.

c) $(C2.2F.14.81)_{16}$ = $(11000010.00101111.00010100.10000001)_2$ = $(194.47.20.129)_{10}$. Thus, belongs to C.

d) Maximum number of VCs over a link = 2^8 = 256.

e) The centralized node could pick any VC number which is free from the set {0, 1, …, 2^8-1}. In this manner, it is not possible that there are fewer VCs in progress than 256 without there being any common free VC number.

Problem 2 Solution

a) 128.96.39.10 & 255.255.255.128 = 128.96.39.0. So next hop is port0
b) 128.96.40.151 & 255.255.255.128 = 128.96.40.0. So next hop is R2
c) R4
d) R3
e) R4

Problem 3 Solution

a) Usually P2P applications broadcast the IP address and port number that they listen on. If the host that they are running is behind a NAT, then the IP address and port number that they broadcast cannot be used by other peers to create connections to them. For example, Bob is running on a machine with IP 192.168.0.1 and listening on port 80. However, all Bob's packets will pass through the NAT, so their source IP and port number will be something different. As a result, Alice is not able to connect to the IP address and port number that Bob broadcasts.

b) When Bob connected to the central server, the central server will see the NATed IP address and port number of the connection that created by Bob, and the server can give this information to Alice. Considering that NAT is bidirectional, if Alice tries to connect to this address, the connection request (i.e. SYN packet) will pass Bob's NAT and reaches Bob.

Problem 4 Solution

The shortest path from B to D is B A C D. The cost of this path is 5.

N'	D(A), p(A)	D(C), p(C)	D(D), p(D)	D(E), p(E)	D(F), p(F)
B	2, B	8, B	∞	10, B	4, B
B,A		4, A	6, A	10, B	4, B
B,A,C			5, C	10, B	4, B
B,A,C,F			5, C	6, F	
B,A,C,F,D				6, F	

Problem 5 Solution

a) The distance vector is shown as

	Destination					
node	A	B	C	D	E	F
A	0	2	2	3	7	6
B	2	0	4	5	6	4
C	2	4	0	1	5	7
D	3	5	1	0	4	6
E	7	6	5	4	0	2
F	6	4	7	6	2	0

b) From its neighbors, nodes A, B, and D. Note that C does not receive distance vectors from nodes E and F, since they are not direct neighbors.

c) C's cost to E via B is c(C, B) + D_B(E) = 8 + 6 = 14
C's cost to E via A is c(C, A) + D_A(E) = 2 + 7 = 9
C's cost to E via D is c(C, D) + D_D(E) = 1 + 4 = 5
Thus, C will route to E via D, since that path through D has minimum cost.

d) From its neighbors, nodes B, D, and F. Note that E does not receive distance vectors from nodes A and C, since they are not direct neighbors.

e) E's cost to B via B is c(E,B) + D_B(B) = 10 + 0 = 10
E's cost to B via D is c(E, D) + D_D(B) = 4 + 5 = 9
E's cost to B via F is c(E, F) + D_F(B) = 2 + 4 = 6
Thus, E will route to B via F, since that path through F has minimum cost.