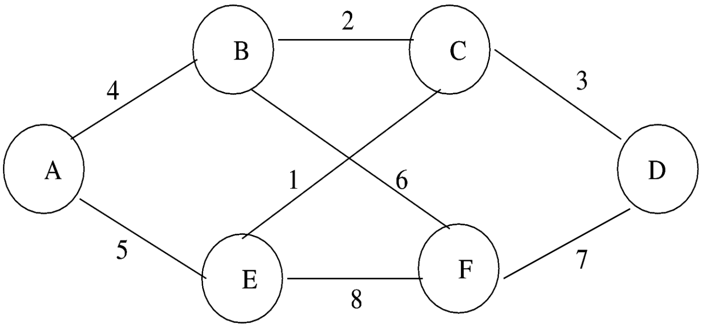
Department of Computer Science and Engg, IIT Bombay
CS 348: Computer Networks Quiz2: 29th Oct 2012, 08:30 – 09:30
Handwritten notes permitted. No photocopies. Weightage: 10%, Max Marks – 30
Answer in the space provided below each question. Extra sheets will not be provided.

Write your Roll Number here: ________________

Q1: Consider the network of routers shown below, running OSPF. Show the steps in the creation of the routing table at router A. **(5 marks)**



Permanent	Temporary
(A,0)	(B,4), (E, 5)
(A,0), (B, 4)	(E, 5), (C, 6, B), (F, 10, B)
(A,0), (B, 4), (E, 5)	(C, 6, B), (F, 10, B), (C, 6, E), (F, 13, E)
(A,0), (B, 4), (E, 5), (C, 6, B); (C, 6, E) instead of (C, 6, B) is also correct	(F, 10, B), (C, 6, E), (F, 13, E), (D, 9, B); Should be (D, 9, E) if (C, 6, E) is chosen
(A,0), (B, 4), (E, 5), (C, 6, B), (D,9 B)	(F, 10, B), (C, 6, E), (F, 13, E), (F, 16, B);
(A,0), (B, 4), (E, 5), (C, 6, B), (D,9 B), (F, 10, B)	-

Q2: Consider a source S using TCP-Tahoe (slow-start and congestion-avoidance mechanisms) to send data to a destination D. Assume that the connection has a constant Round-Trip- Time (RTT), there are no transmission errors and there is no other traffic on the network. Suppose the intial value of ssthresh is 32, the Advertised-Window is constant at 64, and the connection is able to carry 32 packets per RTT (if 33 or more are sent in one RTT, there will be a packet loss). Draw a graph of sending window behaviour (number of packets (cwnd) sent versus time (in units of RTT)), till there are 2 packet losses. **(5 marks)**

This is a straight-forward modification of the graph in slide 39 of
http://www.cse.iitb.ac.in/~sri/cs348/cs348-lec22-25-TCP-2012.pdf

cwnd starts at 1 and goes upto 32 (ssthresh) in 5 RTTs; then increases linearly to 33 packets in the next RTT. Now there is a packet loss.
cwnd is reset to 1, ssthresh is reset to [32/2] = 16; Now cwnd doubles upto 16 and then increases linearly to 33. Now there is a packet loss.
cwnd is reset to 1, ssthresh is reset to [32/2] = 16; Same pattern repeats.

Write your Roll Number here: ________________

Q3: Two hosts, H1 and H2 establish a TCP connection between themselves. H1 used an initial sequence number of 601, while H2 used the initial sequence number of 1550. H1 sends a total of 300 bytes during the connection, and H2 sent 1000 bytes. What is the sequence number and the acknowledgement number of the very last packet sent by H2? **(5 marks)**

Last packet of H2: SEQ no = 2550; ACK no = 902.

Q4: Consider a source using TCP-Reno (fast-retransmit and fast-recovery mechanisms) to send data to a destination. Suppose the RTT of the link is 800 ms and the sender's window size is 8 segments. The sender sends segments at a regular rate of one every 100 ms, and the receiver sends ACKs back at the same rate without delay. A segment is lost, and the receiver sends 3 duplicate ACKs to trigger the fast-retransmit. Suppose the sender waits for ACK of the retransmitted segment before advancing the window, how much total time has the sender lost (as compared to lossless transmission)? **(5 marks)**

Total time lost = 1100 ms.
We wait 300 ms initially to detect the third duplicate ACK, and then one full 800 ms RTT as the sender waits for the ACK of the retransmitted segment.

Suppose pakcet number P is sent at t=0 by the sender's clock. Suppose P is lost.
7 packets – P+1 to P+7 - will be sent at t=100, 200, ..., 700, since the window size is 8.
Sender stops sending at t=800, since window size is 8 and ACK for P has not arrived.
ACK for P+1, P+2 and P+3 will be duplicate ACKs of P-1, and will arrive at t=900, 1000 and 1100.
P is re-transmitted at t=1100. ACK arrives at 1900.
Time lost = 1900-800 = 1100.

Q5: Consider a simple UDP-based protocol for requesting files. The client sends an initial file request and the server answers with the first data packet. Client and server then continue with a stop-and-wait transmission mechanism. Describe a scenario by which a client might request one file but get another. **(5 marks)**

The client sends a request for file X, and immediately crashes. The request arrives at the server.
The client reboots and sends a new request for file Y, using the same port number as for file X. The new request is lost.
The server responds with first data packet of X, answering the only request it has actually seen.
Client receives the data packet and assumes that it is for file Y.

Q6: Suggest a method for marshalling (packing) a linked list while using RPC. **(5 marks)**

This is a straight-forward modification of the example in slide 32 of
http://www.cse.iitb.ac.in/~sri/cs348/cs348-lec29-30-RPC-2012.pdf

In the simplest case, assume a singly linked list of some primitive data types (int/char). Then traverse the list and pack it as: [N, e_1, e_2,.., e_N], where N is the number of items in the list.