1. What are the six flags used in TCP? Why are they needed? (3 points)
   SYN – For connection establishment
   ACK – indicates that the acknowledgement number is valid
   FIN – for termination of connection (I have no more data to send)
   PSH – do not buffer – push the packet to the application
   URG – urgent byte in packet at location indicated in the “urgent” field
   RST – rejection of TCP packet

2. Explain the process of connection establishment in TCP indicating the contents of the flags, sequence number, and acknowledgement fields. (3 points)
   a) Initiator chooses random sequence number and sends SYN packet indicating starting sequence number
   b) Responder chooses random starting sequence number; sends SYN/ACK packet indicating starting sequence number and acknowledgement number (one higher than the initiator's sequence number)
   c) Initiator sends ACK packet with acknowledgement number one higher than the responder's starting sequence number.

3. Explain the need for dynamic window size in TCP? (4 points)
The window size in TCP packets indicates the free space remaining in the buffer. At any time, data bytes received but have not been read by the application will remain in the buffer thus reducing the available space.

4. What is TCP window scaling? Why is it needed? (3 points)
The window size is a 16 bit value. As the maximum window size 64 K the sender can never send more than 64 K bytes before an acknowledgement is received. This is too little for today's Internet speeds as substantially more than 64 K bytes can be sent within the round trip time over fast connections. The window scaling factor indicates the amount by which the window size is scaled. Scaling factor (which can be 1 to 13) of 10 indicates the window size should be scaled by $2^{10}$.

5. What is the silly window syndrome? (3 points)
Silly window syndrome happens when application reads a small number of bytes at a time with long pauses in between. When the buffer is full the TCP layer sends a window size zero advertisement. Application reads one byte and the TCP layer sends a non zero window size advertisement. TCP layer receives a data bytes and buffer is full, triggering a window size advertisement of zero. This sequence keeps repeating for every byte causing substantial overhead for TCP control packets.

6. What is “congestion window size” in TCP? How is it modified? (4 points)
The maximum number of bytes that can be sent by a sender (before an acknowledgement is received from the receiver) is the minimum of the window size advertised by the receiver, and the congestion window size of the sender. Congestion window size is initially set to the MTU size. Thus only one packet (with MTU bytes) can be sent first. If the ACK for the first packet is received in time the congestion window size is doubled. If both packets are acknowledged in time the congestion window size is doubles again. Doubling goes on till a threshold is reached at which point the congestion window size is increased linearly (by one MTU size) instead of exponentially. If at any time a time out occurs the threshold is halved, and the congestion window size is reset to one MTU (and the process of exponential doubling starts all over again).

7. What are the three timers used in TCP? Explain their purpose. (5 points)
   Retransmission timer – started whenever any data packet or SYN packet is sent; ACK should be received before the timer fires (else the packet is retransmitted)
   Persistence times – prevent deadlocks when a non zero window size advertisement (following a zero window size advertisement) is lost. If no packet is received before the time fires the non zero window size advertisement is retransmitted.
   Keep-alive timer – to ping the other side to see if the connection is still alive (in situations where end points prefer to maintain the connection but has currently no data to send).