CS5222 Computer Networks and Internets
Tutorial 4 (Week 4)

1. Suppose we use 8-bit sums instead of 16-bit sums to compute UDP checksum. What's the checksum of three bytes: 01010011, 01010111, 01110100?

2. With the 1's complement of the sum, how does the receiver detect errors? Is it possible that a 1-bit error will go undetected? How about a 2-bit error?

3. Consider distributing a file of F = 15 Gbits to N peers. The server has an upload rate of u_s = 30 Mbps, and each peer has a download rate of d_i = 2 Mbps and an upload rate of u = 700 Kbps. Give the minimum distribution time for N = 10 and N = 1,000 for both client-server distribution and P2P distribution.

4. Suppose that Alice writes a Bittorrent client that doesn't allow other clients to download any data from her system. She claims that, by using this client, she can join a torrent and still receive a complete copy of the shared file. Is Alice's claim possible? Why or why not?

5. Consider a short, 10-meter link with a propagation speed of $300*10^6$ m/sec, over which a sender can transmit at a rate of 150 bits/sec in both directions. Suppose that packets containing data are 100,000 bits long, and packets containing only control (e.g., TCP or HTTP GET request) are 200 bits long. Assume that N parallel connections each get 1/N of the link bandwidth.[1] Now consider the HTTP protocol and suppose that each downloaded object is 100 Kbits long, and that the base HTML file contains 10 referenced objects located at the same host.
    a. Would parallel downloads via 10 parallel instances of non-persistent HTTP make sense in this case? How long does it take to receive all objects?
    b. Now consider persistent HTTP (without parallel connections). Do you expect significant gains over the non-persistent case?

    Note: Keep in mind that you cannot ignore the propagation delay of (small) packets in this question!

[1] Note that here we consider packet switching and *not* circuit switching. Therefore, strictly speaking, the (1/N)-fraction of the bandwidth that each connection "gets" is not reserved, but simply corresponds to the expected transmission rate of that connection if we assume a fair packet forwarding schedule.