Midterm Revision Practice
MSOL CS118 Fall 2013

1. (P3 of Chapter 3 problems in 5th Edition of textbook) UDP and TCP use 1s complement for their checksums. Suppose you have the following three 8-bit bytes: 01010011, 01010100, 01110100. What is the 1s complement of the sum of these 8-bit bytes? (Note that although UDP and TCP use 16-bit words in computing the checksum, for this problem, you are asked to consider 8-bit sums.) Why taking 1s complement of the sum instead of just using the sum? How does the receiver detects error? Is it possible that a 1-bit error goes undetected? How about a 2-bit error?

   Solution:
   Adding the three bytes. Note: *a carryout from the most significant bit needs to be added to the result (wrap around when overflow).*

```
    0 1 0 1 0 0 1 1
+   0 1 0 1 0 1 0 0
-----------------------
    1 0 1 0 0 1 1 1

    1 0 1 0 0 1 1 1
+   0 1 1 1 0 1 0 0
-----------------------
    0 0 0 1 1 1 0 0
```

   1's complement = 1 1 1 0 0 0 1 1.
   To detect errors, the receiver adds the four words (the three original words and the checksum). If the sum contains a zero, the receiver knows there has been an error. All one-bit errors will be detected, but two-bit errors can be undetected (e.g., if the last digit of the first word is converted to a 0 and the last digit of the second word is converted to a 1).

2. For a html file with *n* referenced objects, how many RTTs will be needed if using: (a) Non-persistent HTTP without parallel TCP connections. (b) Non-persistent HTTP with max of *m* parallel TCP connections. (c) Persistent HTTP without pipelining. (d) Persistent HTTP with pipelining.

   Solution:
   (a) (n+1)*2
   (b) if m>=n, 2+2=4; if m<n, 2+[int(n/m)+1]*2.
   (c) 1+1+n=n+2
   (d) 1+1+1=3

3. Describe how TCP congestion control works, i.e. name the phases, trigger events and window size change.

   Solution:
   Slow start -> Congestion avoidance -> 3 duplicated Acks -> Window size cut in half

-> Timeout -> Window size reset to 1