HW 3 CS 222 Winter 2011 Due Weds. January 26 (extension date to Friday Jan. 28)

1. Write a complete explanation (proof) for the following claim made in the discussion of the adversary argument for median finding: After any algorithm has correctly found the position of the median element (by specifying pairs i,j of positions and learning if element i is larger or smaller than element j) then for any position k (not equal to the position of the median element), the algorithm can deduce whether element k is larger or smaller than the median element. Equivalently, if we represent every answer obtained during the running of the algorithm by a directed edge (from the position of larger element to the position of the smaller element), then for any position k, there must be a directed path from k to the position of the median, or to k from the position of the median.

2. It has been suggested that setting $t = \log_2 n - \log_2(\log_2 n)$ is better than setting $t = \log_2 n$ in the Four Russians method for bit matrix multiplication. Following the analysis given in the notes posted on-line, analyze the two choices for t, both for the preprocessing time and for the multiplication time. Is it true that one choice for t is better than the other?

3. Read Section 6.5 of the book on RNA folding. Now suppose that instead of wanting to find the the permitted matching of largest cardinality, we want to *count* the exact *number* of permitted matchings (recall that a single matching is a set of permitted pairs of positions). The counting problem can be solved in $O(n^3)$ time by the following DP:

$N(i, j)$ is defined as the number of permitted matchings involving the positions from i to j inclusive. It includes the empty matching as one of the matchings. For technical reasons, we define $N(j, j) = N(j + 1, j) = 1$. $B(i, k)$ is a binary variable that takes on the value of 1 if the character at position i is allowed to pair with the character at position k, according to rules i) and ii) on page 274. Then the general recurrence is:

$$N(i, j) = N(i + 1, j - 1)$$
$$+ \sum_{i<k\leq j:\texttt{B(i,k) == 1}} [N(i + 1, k - 1) \times N(k + 1, j)]$$
$$+ \sum_{i<k<j:\texttt{B(j,k) == 1}} [N(i + 1, k - 1) \times N(k + 1, j - 1)]$$

Argue that these recurrences give a correct recursive solution for the problem of counting the number of permitted matchings. As before, instead of using the recurrences in a top-down recursive algorithm, we want to use them in a DP solution to the problem. Write out a DP solution to the counting problem, and analyze the worst-case running time of your DP solution.