EECS660 **HW4** **Due:** 10-16-14

Instruction: Read and follow the Homework Policy carefully. You must show all your work clearly for credit. Partial credit will only be given to meaningful answers. You will be graded according to your approach to the problems, mathematical rigor, and quality of your solutions. In designing a greedy algorithm for a specific problem, you must not present the generic greedy algorithm as solution. Your greedy algorithm must be tailored for the given problem.

1. Consider the following recurrence equation:

$$T(1) = 1,$$

$$T(n) = 4T(\frac{n}{2}) + n^2 - n + 1, n = 2^k > 1.$$

   (a) (5) Use the method of domain transformation to transform the given recurrence equation into an IHLRECC with initial condition.
   (b) (10) Use Th. 8 to find a particular solution for the IHLRECC and then solve it. Compute T(n).
   (c) (10) Use Th. 9 to construct the characteristic equation for the IHLRECC and then solve it. Compute T(n).

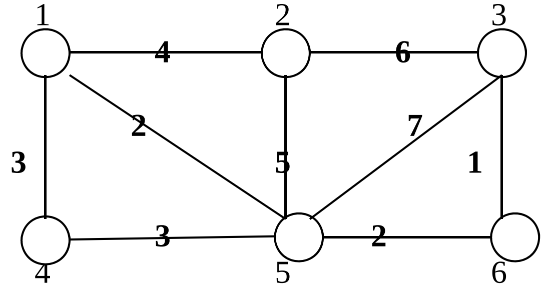
2. (15) Prove by using examples of no more than 5 turkeys that none of the three greedy algorithms used for solving the Turkey Problem (Knapsack Problem) is optimal. You must justify your answer clearly for credit. (Try to use as few turkeys to yield as bad a solution as possible in your examples.)

3. (10) Prove by using example that the greedy method of making change using only pennies, dimes, quarters, and half-dollar may not always yield an optimal solution (See notes). You must justify your answer clearly for credit.

4. (20) Given a sequence n matrices $M_1, M_2, \ldots, M_n$ with dimensions $<d_0, d_1, d_2, d_3, \ldots, d_{n-1}, d_n>$, where matrix M_i is of dimension $d_{i-1} \times d_i$. Design and analyze a greedy algorithm to multiply this sequence of matrices together so that the total number of multiplications required is minimized? Prove or disprove that your greedy algorithm will always generate an optimal ordering for multiplications.

5. (15) Given the following weighted graph G = (V,E). Compute a min spanning tree for G using Kruskal's algorithm. **Remark:** You must show all iteration, list the edges, including those rejected by your algorithm, in their order of selection for credit.

G:



6. (15) Starting at vertex 1, compute a min spanning tree for the graph G in Problem 5 using Prim's labeling algorithm. **Remark:** You must show all iteration clearly for credit.

Remark: Turn in Problems 1, 2, and 3 for grading.