MCS 401 Final: DH 210, 8:00-10:00, Tue May 5, 2009

The following problems are suggested for practice before the final. In addition review homework problems in the related chapters. The exam will cover

- Dynamic programing, greedy algorithms (concepts only, no memorization of examples is necessary).
- Fibonacci heaps (basic concepts only).
- Graph algorithms: BFS, DFS, their applications (classification of edges, topological sort, SCCs, etc.).
- Minimal spanning trees: Kruskal, Prim.
- Shortest paths algorithms: single-source (Bellman-Ford, Dijkstra), all pairs ($\odot$-algorithms, Floyd-Warshall).

1. Given n integer denominations of coins $1 = v_1 < v_2 < \ldots < v_n$ and an integer amount of money C determine the minimum number $m(C)$ of coins needed to make change for C.
   (a) Describe the optimal substructure. (Hint: given an optimal combination of coins consider removing one of them.)
   (b) Using the described substructure, write a recurrence for $m(C)$.
   (c) Assuming the values $m(x)$ for all $x = 1, \ldots, C$ have been computed, describe how to construct a combination of coins $(a_1, \ldots, a_n)$ such that $\Sigma_{i=1}^{n} a_i v_i = C$.

2. A run of depth-first search on a directed graph with vertices $A, B, \ldots, J$ yields the following discovery/finish times:

vertex	A	B	C	D	E	F	G	H	I	J
d	1	2	10	12	3	13	17	4	6	14
f	20	9	11	19	8	16	18	5	7	15

   (a) Sketch the predecessor forest created by this run of DFS.
   (b) Could the graph have an edge from J to D? yes no
   If so, then what kind? tree, forward, back, cross (underline all possibilities)
   (c) Could the graph have an edge from E to D? yes no
   If so, then what kind? tree, forward, back, cross (underline all possibilities)
   (d) What is the maximum number of edges starting at I that could be present in the graph?

3. Describe an $O(n)$-time algorithm that, given $x_1 < x_2 < \cdots < x_n$ on the real line, determines a smallest set of unit-length closed intervals the union of which contains all x_i. Prove your algorithm is correct.

4. Given a modification of Kruskal's algorithm to solve the following problem. Let $G = (V, E)$ be a connected, undirected graphs with weighted edges, and S be a subset of E containing no cycle. Find a spanning tree of G containing S, such that its weight is as small as possible. What is the running time of your algorithm?

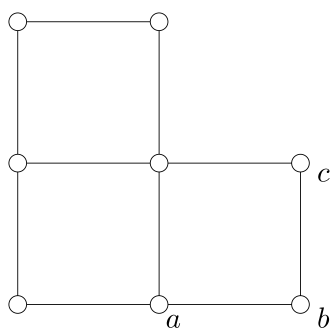
5. Consider a directed graph $G = (V, E)$.

   (a) Give a definition of a *strongly connected component* of G.

   (b) Give a definition of the *component graph* G^{SCC}.

   (c) Prove that G^{SCC} is acyclic.

6. (a) Does the following graph have a spanning tree where a, b, c are leaves? Explain.



   (b) Given an undirected graph $G = (V, E)$ and a subset $U \subset V$, give an $O(E)$-time algorithm to determine whether G has a spanning tree such that every $u \in U$ is a leaf.

7. One way of reconstructing shortest paths in the Floyd-Warshall algorithm is to use values $\phi_{ij}^{(k)}$ with $i, j, k \in \{1, \cdots, n\}$, where $\phi_{ij}^{(k)}$ is the largest label of an intermediate vertex of a shortest path from i to j with labels of the intermediate vertices in $\{1, \ldots, k\}$.

   Modify FLOYD-WARSHALL to compute values of $\phi_{ij}^{(k)}$.