MCS 401 Final: BH 208, 10:30-12:30, Thu, Dec 11, 2008

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).
- Graph algorithms: BFS, DFS, their applications.
- Minimal spanning trees: Kruskal, Prim.
- Shortest paths algorithms: single-source (Bellman-Ford, Dijkstra), all pairs (Floyd-Warshall).
- Maximum flow: flow networks, Ford-Fulkerson method.

1. 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?

2. 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.

3. Let $G = (V, E)$ be a directed graph with $|V| = n$ and $|E| = m$. Suppose that each vertex $u \in V$ is labeled with a unique integer $L(u)$ from the set $\{1, 2, \ldots, n\}$. For each $u \in V$, let $R(u) = \{v \in V : u \leadsto v\}$ be the set of the vertices that can be reached by a directed path starting from u. Define $\min(u)$ to be the vertex in $R(u)$ whose label is minimum. Give an $O(n + m)$-time algorithm that computes $\min(u)$ for all vertices $u \in V$.

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. Sketch an $O(V)$-time algorithm that, given a directed graph $G = (V, E)$ and $u, v \in V$, will determine whether there is a path $u \rightsquigarrow v$.

6. 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.

7. The following flow network has edges labeled with pairs of numbers representing flow/capacity. Draw the *residual network* and give an example of an *augmenting path*. What additional flow does this path provide?