CMPT 307-d1 Final
April 14, 2007

Answer questions worth 90 points. Questions 1 and 7 must be answered.

1. (15 points) Let $T(n)$ denote the running time of a procedure with input parameter n. For each of the following procedures find the order of T (that is, find a function $f(n)$ such that $T \in O(f(n))$.

   (a) **procedure** Foo(**integer** n)
       for i **from** 1 **to** n **do**
           $x \leftarrow n$
           while $x > 0$ **do**
               $x \leftarrow x - i$

   (b) First define $T(n)$ recursively, and then find $f(n)$.

   procedure $MAXMIN(S)$
       if $|S| = 2$ **then** { S has two elements a and b}
         return (MAX(a, b)**, MIN**(a, b)**)**
       else
         divide S into two subsets S_1 and S_2, each with half the elements
         $(max1, min1) \leftarrow MAXMIN(S_1)$
         $(max2, min2) \leftarrow MAXMIN(S_2)$
         return (MAX($max1, max2$**),MIN(**$min1, min2$**))**

   You can assume that the number of elements in S is a perfect power of 2. The procedures **MAX**(a, b) and **MIN**(a, b) determine the maximum and the minimum elements of a and b respectively.

   (c) Define $T(n)$ recursively, and then find $f(n)$.

   procedure $Tow(n)$
         if $n = 1$ **return** n
         else return $Tow(n-1) + Tow(n-1)$

2. (10 points) Consider the abstract operation $SUBSET$, where $SUBSET(S, T)$ returns **true** if S is a subset of T aand **false** otherwise. Explain how to implement the abstract operation for each of the following data structures: unordered lists, ordered lists. Express the running times in terms of the sizes of S and T. Does it help if the sets S and T are stored in binary search trees? (Try to avoid a sequence of *Lookups*.)

3. (10 points) Write a function $FastExp$ such that $FastExp(x, n) = x^n$ for any real number x and for any positive integer n, using atmost $2\log_2 n$ multiplications.

4. The **transpose** of a directed graph $G = (V, E)$ is the graph $G^T = (V, E^T)$ where $E^T = \{(v, u) \text{ if } (u, v) \in E\}$. Thus, G^T is G with all the edges reversed. Describe an efficient algorithm for computing G^T from G when G is represented by an adjacency list. Analyze the running times of the algorithm.

5. (10 points) In the Acyclic Subgraph Problem, we are given a directed graph $G = (V, E)$ and we are asked to pick a subset E' of E such that $G = (V, E')$ has no cycles. Consider the following algorithm: Without loss of generality assume that $V = \{1, ..., n\}$. Let $S_1 = \{(u, v) \in E : u < v\}$ and $S_2 = \{(u, v) \in E : u > v\}$. The output is the larger of the two sets S_1 and S_2. Show that neither S_1 nor S_2 contains cycles.

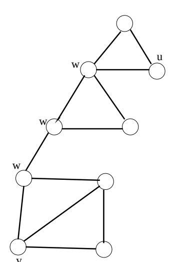
6. (15 points) Define the Longest Common Subsequence (LCS) problem. Give a memoized version of LCS problem of two strings $X =< x_1, x_2, \ldots, x_m >$ and $Y =< y_1, y_2, \ldots, y_n >$ that runs in $O(mn)$ time. What is the storage space complexity?

7. (a) Describe a dynamic programming algorithm to optimally make change of 16^c for the denomination set $\{1^c, 5^c, 10^c, 12^c\}$.
(b) Describe a greedy algorithm for the above problem. Is greedy optimal for the given denomination set?
(c) Show that the greedy approach is optimal for the denomination set $\{1, 2^1, 2^2, \ldots, 2^k\}$. (Hint: You need to show first that to make change of atleast 2^k, the optimal solution must contain a 2^k coin.)

8. (15 points) Consider the following graph.

   (a) Apply the depth first search on the graph given below. Determine the timestamps for each vertex (i.e. when the vertex was first visited and when the vertex was last visited). Classify the edges into tree edges, back edges.

   (b) A vertex w in a graph G is called a cut vertex if there exist atleast a pair of vertices u and v such that every path between u and v contains the vertex w. In the figure each of the vertices labeled w is a cut vertex. Design an algorithm to identify the cut vertices once the timestamps for each vertex of the graph are known. (The locations of the black edges are important.)



9. (20 points) Are the following statements right or wrong? For each statement, prove it if it is correct; construct a counterexample if it is wrong.

   (a) In an undirected graph with positive edge distances, the shortest edge belongs to every single source shortest paths of the graph.

   (b) If all edges in a graph have different lengths, then the shortest path tree from any source node to all other nodes is unique.

(c) Partition an undirected graph into two parts and construct the minimum spanning tree of each part. Then connect the two parts by the shortest cross edge. The resulting tree is a minimum spanning tree for the entire graph.

(d) The path between a pair of vertices in a minimum spanning tree of an undirected graph with distinct edge costs is a minimum-cost path.

(e) Given a connected, undirected graph with distinct edge weights, then the edge e with the second smallest weight is included in the minimum spanning tree.

10. (15 points) The matrix shown below is the adjacency matrix of an undirected graph. State Prim's agorithm in terms of the matrix and then apply the algorithm to the matrix.

v_1	v_2	v_3	v_4	v_5	v_6	v_7	
v_1	0	4	5	12	∞	∞	∞
v_2	4	0	3	∞	1	∞	∞
v_3	5	3	0	1	∞	∞	13
v_4	12	∞	1	0	∞	11	∞
v_5	∞	1	∞	∞	0	∞	9
v_6	∞	∞	∞	11	∞	0	8
v_7	∞	∞	13	∞	9	8	0