George Washington University
Department of Computer Science

CS212 - Practice Problems for Final Examination

1. Give a dynamic programming algorithm for all-to-all shortest paths problem.

2. Give a definition of the *optimal binary search tree* discussed in class. Give a dynamic programming algorithm to construct an optimal binary search tree.

3. Describe a dynamic programming algorithm for the Traveling Salesperson problem. Analyze the worst-case time complexity.

4. Consider the algorithm to compute all biconnected components of a given graph.

(a) Give a definition of the following: articulation point, biconnected, biconnected subgraph, maximal biconnected subgraph, biconnected component, low point.

(b) Apply the algorithm discussed in class to find all biconnected components of the following graph (for any arbitrary given graph).

(c) Describe a necessary and sufficient condition for a vertex to be an articulation point. (Justify your answer.)

5. Given a sequence of n real numbers $A(1), \cdots, A(n)$, give a dynamic programming algorithm to determine a contiguous subsequence $A(i), \cdots, A(j)$ for which the sum of elements in the subsequence is maximized. Analyze the time complexity of your algorithm.

6. You are given n types of coin denominations of values $v(1) < v(2) < \cdots < v(n)$ (all integers). Assume $v(1) = 1$, so you can always make change for any amount of money C. Give a dynamic programming algorithm which makes change for a given amount of money C with as few coins as possible. Analyze the time complexity of your algorithm.

7. Given a sequence of n real numbers $A(1), \cdots, A(n)$, Give a dynamic programming algorithm to determine a subsequence (not necessarily contiguous) of maximum length in which the values in the subsequence form a strictly increasing sequence. Analyze the time complexity of your algorithm.

8. Consider a 2-D map with a horizontal river passing through its center. There are n cities on the southern bank with x-coordinates $a(1), \cdots, a(n)$ and n cities on the northern bank with x-coordinates $b(1), \cdots, b(n)$. You want to connect as many north-south pairs of cities as possible with bridges such that no two bridges cross. When connecting cities, you can only connect city i on the northern bank to city i on the southern bank. Give a dynamic programming algorithm to solve this problem and analyze the time complexity of your algorithm.

9. You have a set of n integers each in the range $0, \cdots, K$. Partition these integers into two subsets such that you minimize $|S_1 - S_2|$, where S_1 and S_2 denote the sums of the elements in each of the two subsets. Give a dynamic programming algorithm to solve this problem and analyze the time complexity of your algorithm.

10. Given two text strings A of length n and B of length m, you want to transform A into B with a minimum number of operations of the following types: delete a character from A, insert a character into A, or change some character in A into a new character. The minimal number of such operations required to transform A into B is called the *edit distance* between A and B. Give a dynamic programming algorithm to solve this problem and analyze the time complexity of your algorithm.

11. You are given a boolean expression consisting of a string of the symbols 'true', 'false', 'and', 'or', and 'xor'. Count the number of ways to parenthesize the expression such that it will evaluate to true. For example, there is only 1 way to parenthesize 'true and false xor true' such that it evaluates to true. Give a dynamic programming algorithm to solve this problem and analyze the time complexity of your algorithm.

12. You are given an ordered sequence of n cities, and the distances between every pair of cities. You must partition the cities into two subsequences (not necessarily contiguous) such that person A visits all cities in the first subsequence (in order), person B visits all cities in the second subsequence (in order), and such that the sum of the total distances traveled by A and B is minimized. Assume that person A and person B start initially at the first city in their respective subsequences. Give a dynamic programming algorithm to solve this problem and analyze the time complexity of your algorithm.

13. You have n_1 items of size s_1, n_2 items of size s_2, and n_3 items of size s_3. You'd like to pack all of these items into bins each of capacity C, such that the total number of bins used is minimized. Give a dynamic programming algorithm to solve this problem and analyze the time complexity of your algorithm.

14. Characterize all undirected graphs that contain a vertex v such that there exists a DFS tree rooted at v that is identical to a BFS tree rooted at v. Note that two spanning trees are identical if they contain the same set of edges.

15. Given a connected undirected graph G and three edges (u_1, v_1), (u_2, v_2), (u_3, v_3) of $E(G)$, design an $O(|E|)$ time algorithm to determine whether there exists a cycle in G that contains both (u_1, v_1) and (u_2, v_2), but does not contain (u_3, v_3).

16. An undirected graph G is called *complete* if for every pair of vertices $u, v \in V(G)$, there is an edge $(u, v) \in E(G)$. A subgraph $H \subseteq G$ is called a *clique* of G if if for every pair of vertices $u, v \in V(H)$, there is an edge $(u, v) \in E(H)$.

(a) Design a backtracking algorithm to find all cliques of size k (i.e., with k vertices), where k is an arbitrary given integer, $1 \leq k \leq n$.

(b) Now, assume that each node v is assigned a positive weight $w(v)$. Design a branch-and-bound algorithm to find a clique in G such that the sum of the weights of vertices in the clique is maximized. (We call such a subgraph a *maximum clique.*) For example, consider a graph G with $V(G) = \{a, b, c, d, e\}$, and $E(G) = \{(a, b),\ (a, c),\ (b, c),\ (b, d),\ (b, e),\ (c, d),\ (c, e),\ (d, e)\}$, where $w(a) = 5$, $w(b) = 3$, $w(c) = 2$, $w(d) = 2$ and $w(e) = 1$. Then, the subgraph H with $V(H) = \{a, b, c\}$ and $E(H) = \{(a, b), (b, c), (c, a)\}$ is a maximum clique of G.