1) An undirected graph is k-connected if, for every pair of vertices v and w, there are k paths between *v* and *w* such that no vertex (except *v* and *w*) appears in more than one path. Biconnected means 2-connected. Write a linear time (in number of vertices and edges) algorithm to find the 3-connected components of a graph.

2) Suppose that we have 2*n* elements $<a_1,a_2,\ldots a_{2n}>$ and wish to partition them into the *n* smallest and the *n* largest. Prove that we can do this in constant additional depth after sorting $<a_1,a_2,\ldots a_n>$ and $<a_{n+1}, a_{n+2},\ldots a_{2n}>$.

3) Design a dynamic programming algorithm for the change-making problem: Given an amount n and unlimited quantities of coins of each of the denominations $d_1, d_2, \ldots d_m$, find the smallest number of coins that add up to *n* or justify that the problem does not have a solution.

4) Show that for an undirected graph G (V, E), 3-coloring is NP complete. *A valid coloring of G is an assignment of colors to the vertices such that each vertex is assigned one color and no two adjacent vertices have the same color.*

5) Give an efficient Greedy algorithm that finds an optimal vertex cover for a tree.

6) Given two vertices *s* and *t* of an undirected graph, show how to use network flow to determine whether there are at least two paths between s and t that do not share any common intermediate edge (the paths are allowed to share common vertices).

7) Textbook, Chapter 3, Problem Number 11, page 111.

8) Textbook, Chapter 4, Problem Number 12, page 193.

9) Textbook, Chapter 6, Problem Number 10, page 321.

10) Textbook, Chapter 7, Solved Exercise 2, pages 412, 413.

11) Textbook, Chapter 7, Problem Number 12, page 420.

12) Textbook, Chapter 8, Solved Exercise 2, pages 502-505.