1. (15) Consider the union-find problem on a set of n elements $\{1, 2, \ldots, n\}$. Initially each element is a subset by itself. There are two types of operations on these subsets of S.

   (a) The *find* operation $F(x)$ returns the subset to which the element x belongs.

   (b) The *union* operation $U(u, v)$ makes the two subsets u and v into one subset.

   Note that each subset has a unique name, and the name cannot be changed. Suppose that each subset is represented by a tree. Each vertex has a pointer points to its parent, except the root whose pointer points to itself.

   (a) (5) Assume that $n = 7$, and each subset is named by the smallest element in that subset. Show the subsets by drawing the forest after each of the following sequence of union operations:
   $U(1, 2), U(3, 4), U(1, 3), U(5, 6), U(1, 5)$.

   (b) (10) Design efficient algorithms for the union and find operations so that each find operation can be done in $O(\log n)$ time and the union can done in $O(1)$ time.

2. (15) A bipartite graph is a graph whose vertex set V can be partitioned into two subsets V_1 and V_2, such that no edges connecting vertices in the same partition. A k-cube is a graph whose vertex set is the set of binary strings of length k. Two vertexes u and v are connected if they differs in only 1 bit.

   (a) (5) Draw n-cube for $0 < n \leq 3$.

   (b) (10) Show that k-cubes are bipartite graphs for any $k > 0$.

3. (20) An Euler cycle is a closed walk which traverses each edge exactly once. An Euler path is a walk which traverses each edge exactly once. Let G be a connected undirected graph.

   (a) Show that G has an Euler cycle if and only if the degree of every vertex is even.

   (b) Design a linear time algorithm for finding an Euler cycle for the graph G.

4. (10) Given a weighted graph $G = (V, E, w)$. Suppose that we want to find a spanning tree with maximum weight, instead of a minimum one. Design an algorithm for finding a maximum spanning tree, and justify that your algorithm is correct.

5. (20) Let $G = (V, E, w)$ be a connected weighted graph. Suppose that Dijkstra's algorithm is used to solve the single source shortest path problem on the weighted graph G.

   (a) Give an example to show that Dijkstra's algorithm does not work for graphs with negative weights.

   (b) Can we first add a constant value to the weight of each edge to make all weights non-negative, then run the Dijkstra's algorithm to compute shortest paths? Justify your answer.

6. (10) Let $G = (V, E, w)$ be a connected weighted undirected graph. Given a vertex $s \in V$ and a shortest path tree T_s with respect to the source s, design a linear time algorithm for checking whether the shortest path tree T_s is correct or not.

7. (20) Let n be a product of two large primes p and q. Design a polynomial time nondeterministic algorithm for finding a factor of n, that is, p or q.

8. (20) The $\{0, 1\}$-*knapsack problem* is defined as:

   Given a set of $K = \{(w_i, v_i)\}$, $1 \leq i \leq n$, and a capacity of the knapsack S, find an n-dimensional vector $U = (u_1,\ u_2,\ \ldots,\ u_n)$ in $\{0, 1\}^n$ such that $W = \sum_{i=1}^{n} u_i \cdot w_i \leq S$ and the value of $P = \sum_{i=1}^{n} u_i \cdot v_i$ is maximized.

   The *subset sum problem* is defined as:

   Given a sequence of n integers $X = x_1, x_2, \ldots, x_n$, and an integer S. Fine a subsequence X' of X such that the sum of the integers in X' is S, or report that there are no such subsequences.

   Show that the subset sum problem can be reduced to the $\{0, 1\}$-knapsack problem.