1. Explain each of the following terms. (20%)
   (a) topological ordering
   (b) sum of subset problem
   (c) hill climbing method
   (d) transitive closure of a graph
   (e) closest pair problem

2. Given an undirected graph, what are the Euler cycle and Hamiltonian cycle problems? Please point out their main difference. (10%)

3. (a) Give an algorithm for solving the shortest path (from single source) problem of an undirected graph. Give an example to illustrate your algorithm. (10%)
   (b) Please derive the time complexity of your algorithm. (5%)

4. (a) What is the definition of the 2D ranking problem? Give an example to illustrate your explanation. (5%)
   (b) Give a divide-and-conquer algorithm for solving the problem. Please derive the time complexity of your algorithm. (10%)

5. (a) Give the definition of the longest common subsequence (LCS) problem. (5%)
   (b) Design an algorithm to solve the LCS (length) problem. (5%)

6. Let $S_n = \{1, 2, \cdots, n\}$. An $m$-combination of $S_n$ is obtained by selecting $m$ distinct integers out of the $n$ integers and it is represented with lexicographic order. For example, both combinations (2 4 1) and (4 1 2) are the same, and they are represented with (1 2 4). Now let $x = (x_1 x_2 \cdots x_m)$ and $y = (y_1 y_2 \cdots y_m)$ be two $m$-combinations of $S_n$. We say that $x$ precedes $y$ in lexicographic order if there exists an $i$, $1 \leq i \leq m$, such that $x_j = y_j$ for all $j < i$ and $x_i < y_i$. And the rank of an $m$-combination $c$, denoted as $r(c)$, is the number of combinations before $c$ in the lexicographical order. For example, all 3-combinations of $S_4$ in lexicographical order are (1 2 3), (1 2 4), (1 3 4), (2 3 4). Thus $r(1 2 3) = 0$, $r(1 2 4) = 1$, $r(1 3 4) = 2$, and $r(2 3 4) = 3$. Answer the following questions for the 5-combinations of $S_{10}$.
   (a) What is the next one of (2 3 5 9 10)? (5%)
(b) What is \( r(3 4 6 8 9) \)? Explain how do you calculate? (5%)

(c) Which combination has the rank 178? Explain how do you calculate? (5%)

7. Given a set \( S \) of numbers, you are asked to find whether there exist four numbers \( a, b, c, d \in S \) such that \( a + b + c = d \). For solving this problem, a trivial method is to examine all 4-number combinations, which requires \( O(n^4) \) time where \( |S| = n \). However, this method is not efficient enough. Please design a more efficient algorithm to solve this problem and analyze the time complexity. Note that if your algorithm needs less than \( O(n^3 \log n) \) time, you will get higher score. (15%)