**Dept. of Computer Science and Engineering, undergraduate**
**National Sun Yat-sen University**
**Advanced Programming and Practice - Final Exam., June 19, 2014, 9:10–12:00**

1. Explain each of the following terms. (20%)

   (a) *Euler cycle*

   (b) *longest common subsequence*

   (c) *transitive closure of a graph*

   (d) *optimal binary search tree problem*

   (e) *branch and bound*

2. (a) What is the *topological ordering* in an acyclic directed graph? (5%)

   (b) Present an algorithm for solving the above problem. And analyze the time complexity of your algorithm. (10%)

3. The problem of *job sequencing with deadlines* is described as follows. We are given $n$ jobs, numbered as $1, 2, 3, \cdots, n$, where each job $i$ has a deadline $d_i$ and a profit $p_i$. We need one unit of time to do each job and we can do at most one job each time. We can earn the profit $p_i$ if job $i$ is completed by its deadline. For example, suppose the deadline of job $i$ is 3. Then, if we do job $i$ at time $[0, 1]$, $[1, 2]$, or $[2, 3]$, we can earn profit $p_i$. The problem is to select a subset of jobs with maximum profit, but without deadline violation.

   (a) Suppose that we are given 5 jobs with deadlines $\{3, 1, 2, 3, 2\}$ and profits $\{5, 10, 20, 1, 15\}$. What job numbers are selected? How much can we earn? (5%)

   (b) Present an algorithm for solving the above problem. And analyze the time complexity of your algorithm. (10%)

4. (a) Given $n$ points on the 2D plane, what is the *2D ranking* problem? Please give an example to describe your answer. (5%)

   (b) Present a *divide-and-conquer* algorithm for solving the above problem. And analyze the time complexity of your algorithm. (10%)

5. In the solution tree searching strategy, explain each of the following and give the data structure used by each. (15%)

   (a) *depth-first search*

   (b) *breadth-first search*

   (c) *best-first search*

6. Let $S_n = \{1, 2, \cdots, n\}$. An $m$-permutation of $S_n$ is obtained by selecting $m$ distinct integers out of the $n$ integers and arranging them in some order. Now let $x = (x_1\ x_2\ \cdots\ x_m)$ and $y = (y_1\ y_2\ \cdots\ y_m)$ be two $m$-permutations 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 $m$-permutation $p$, denoted as $r(p)$, is the number of permutaions before $p$ in the lexicographical order. For example, all 3-permutations of $S_4$ in lexicographical order are (1 2 3), (1 2 4), (1 3 2), (1 3 4), (1 4 2), (1 4 3), (2 1 3), $\cdots$, (4 3 2). And $r(1\ 2\ 3) = 0$, $r(1\ 4\ 2) = 4$, $r(4\ 3\ 2) = 23$ and etc. Answer the following questions for the 4-permutations of $S_7$.

   (a) What is the next one of (4 3 7 6)? (2%)

   (b) What is $r(5\ 2\ 3\ 6)$? Explain your way of calculation. (4%)

   (c) Which permutation has the rank 329? Explain your way of calculation. (4%)

7. Given a sequence of integers $a_1, a_2, \cdots, a_n$ (positive, zero, or negative), the *maximum segment* problem is to find a segment $a_i, a_{i+1}, \cdots, a_j$, $1 \leq i \leq j \leq n$, such that $a_i + a_{i+1} + \cdots + a_j$ has the maximum value. In the given sequence, at least one integer is positive. Please design an algorithm for solving the problem with linear time complexity. Note that you will get no point if your algorithm is not of linear time. (10%)