Dept. of Computer Science and Engineering, National Sun Yat-sen Univ.
First Semester of 2011 PhD Qualifying Exam Computer Algorithms

1. Explain each of the following terms. (12%)

   (a) *NP, NP-complete*

   (b) *convex hull problem*

   (c) *branch and bound*

2. It is well-known that based on the comparison operations, the *lower bound* of the *sorting* problem is $\Omega(n \log n)$, where n is the input size. Prove that the lower bound of the *Euclidean minimal spanning tree* problem is also $\Omega(n \log n)$ by problem transformation. (13%)

3. In the solution searching strategy, there are two basic ways to visit solution nodes: *depth-first search* and *breadth-first search*.

   (a) What data structure should be used in depth-first search? Why? (5%)

   (b) What data structure should be used in breadth-first search? Why? (5%)

   (c) What is the *best-first search* scheme? (5%)

4. Rewrite the *sorting* problem to be a decision version. (15%)

5. Suppose we obtain the following recurrence formula of time complexity for solving some problem:

$$T(n) = \begin{cases} b & \text{if } n \leq 2 \\ 2T(\frac{n}{2}) + cn^2 & \text{if } n > 2, \end{cases}$$

   where n is the input size of the solved problem. Please derive the time complexity and represent it with O notation. (15%)

6. Design an algorithm to determine whether a graph $G = (V,E)$ is 2-colorable. Your algorithm should be done in $O(n+m)$ time, where $n = |V|$ and $m = |E|$. Give an example to illustrate your algorithm and explain why your algorithm can be performed in $O(n+m)$ time. (15%)

7. (a) For any arbitrary four numbers a, b, c and d, how do you find the smallest and second smallest numbers with at most 4 comparisons in the worst case? (5%)

   (b) For any arbitrary 16 numbers, design an algorithm for finding the smallest and second smallest numbers with at most 18 comparisons in the worst case. Analyze the number of required comparisons. (10%)