科目：演算法

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

    (a) NP

    (b) NP-complete

    (c) 1-center problem

    (d) minimum spanning tree

    (e) AVL tree

2. (a) Suppose that an $n \times n$ matrix is symmetric. How many entries of the matrix, in terms of n, need be stored when it is used? (10%)

    (b) In an $n \times n$ matrix A, each $A_{ij}=0$ if $|i-j| \geq 3$. What is the maximum number of nonzero elements in the matrix? (10%)

3. (a) Give the definition of the *longest common subsequence* (LCS) problem. (5%)

    (b) Design a *dynamic programming* method for calculating the LCS length. (10%)

4. In the self-organizing sequential search heuristics, what are the *transpose heuristics*, *move-to-front heuristics* and *count heuristics*? (15%)

5. Design an algorithm to find both the *minimum* and the *maximum* of n elements with at most $\lceil 3n/2 \rceil$ comparisons. (15%)

6. Prove that the *partition* decision problem polynomially reduces to the *bin packing* decision problem. (15%)