誠 Honesty 樸 Simplicity

國立台灣海洋大學資訊工程學系博士班

96學年度第一學期博士班資格考命題卷（筆試）

博 Fraternity 毅 Perseverance

科目	演算法	命題教授	吳宗杉老師 翁世光老師	日期	97/01/09

1. Write an algorithm to find the median of an unsorted array of n elements with time $\Theta(n)$ in average case (8%). Analyze the complexity (average case) of the algorithm (7%). Distribute $O(N)$ $T(N) = T(N/2) + N$

2. Briefly answer the following questions: (15%)
   (a) Main recurrence theorem
   (b) Supremum
   (c) Optimal substructure property

3. Compare dynamic programming with divide-and-conquer technique. (10%)

4. For the text searching problem, it can be solved by brute force sequential searching in time $O(|p| * |t|)$, where $|p|$ and $|t|$ stand for the length of pattern p and text t, respectively. The Knuth-Morris-Pratt algorithm solves the pattern-matching problem in time $O(|p| + |t|)$. Demonstrate how the Knuth-Morris-Pratt works with pattern p = "pappar" and text t = "pappappapparrassanuaragh". (10%)

5. Prove that the maximum height of a red-black tree with N nodes is O(2logN) while the minimum height is O(logN). (Hint: consider the maximum and minimum heights of 2-3-4 trees.) (10%)

6. When performing hashing by using a hash function based on division method, we have to select the radix base **d** and the hash table size **M** carefully. Why should **d** and **M** be relatively prime ? (Explain the reasons according to the space spanned by the terms: i***d** % **M** and **d**^{i} % **M.**) (15%)

$$h(x) = (a_i d^i + a_{i-1} d^{i-1} + \ldots + a_2 d^2 + a_1 d + a_0)\%M$$

7. Let G=(V, E) be a weighted graph, and G is connected. Select vertex V0 as the source and construct a shortest path tree of G. Assume the tree is Tshort . Then construct a minimum spanning tree of G and assume the minimum spanning tree is Tmin. Prove that the total edge weights of Tmin are always less than or equal to that of Tshort. Please give an example graph in which both trees have the same total weights. (10%)

8. Let A[N][N] be a two dimensional array with N*N integers. Design a sorting

algorithm with **O(N^{2}longN)** steps to sort **A[][]** such that all elements of **A[][]** are sorted in vertical, horizontal and diagonal directions. That is **A[0][0]** is the smallest element and **A[N-1][N-1]** the largest element. You cannot copy **A[][]** to an 1-D array, perform the sorting, and copy it back. Instead, your algorithm have to sort **A[][]** in place. (Hint: use quick-sort. However, sort A[][] along vertical and horizontal directions alternatively. Otherwise, you can enumerate A[][] in a zig-zag manner during the partition stage and split A[][] into 4 blocks.) (15%)