國立交通大學試題紙

九十七學年度第二次
博士班資格考

科目：演算法(A)　　日期： 98 年 7 月 23 日　第 1 頁 共 3 頁

請"✓"明　✓不可看書　可看書

* 請將答案依題號順序寫入答案卷

答題時字跡需工整，否則不予計分。Write your answers legibly; otherwise you will get zero score.

1. (16%) Rank the following functions by order of growth; that is, find an arrangement $g_1, g_2, \ldots, g_n$, of functions satisfying $g_1 = \Omega(g_2)$, $g_2 = \Omega(g_3)$, …, $g_{n-1} = \Omega(g_n)$. Partition your list into equivalence classes such that $f(n)$ and $g(n)$ are in the same class if and only if $f(n) = \Theta(g(n))$. (Note: $\log n = \log_{10} n$, $\lg n = \lg_2 n$, $\ln n = \log_e n$, $\log_e 2 = 0.693$ and $\log_{10} 2 = 0.301$).

(a) $(\lg n)^{\lg n}$	(b) $n(\lg n)/(\lg\lg n)^3$
(c) $(n)^{1/\lg n}$	(d) $n^{\lg\lg n}$
(e) 1.1^n	(f) 1.01^{10n}
(g) $2n\times(2n-2)\times\ldots\times 2$	(h) n^n
(i) $\log(n!)$	(j) $\log(n^n)$
(k) $n^2 + n^2/2 + n^2/4 + \ldots + 1$	(l) $1^3 n + 2^3(n/2) + 3^3(n/4) + \ldots + (\ln n)^3$
(m) $8^{\lg n}$	(n) $16^{\ln n}$
(o) $n + n/2 + n/3 \ldots + n/n$	(p) $1\times(n-1) + 2\times(n-2) + 3\times(n-3) + \ldots + (n-1)\times 1$

2. (8%) Let $T(n) = \Theta(f(n))$. Assume that $T(n)$ is constant for sufficiently small n. Derive $f(n)$ in the simplest formula for each of the following $T(n)$.

$$T(n) = 2T(n/2) + n\log n.$$
$$T(n) = 2T(n/2) + n.$$
$$T(n) = 2T(n/2) + n/\log n.$$
$$T(n) = 2T(n/2) + n/\log^2 n.$$

3. (10%) The problem of building max-heap is: given n elements, build a max-heap containing the n elements. Design a linear-time algorithm for this problem and prove that its time complexity is O(n).

4. (16%) (a) Briefly describe your algorithm of finding a minimum parenthesization of a matrix-chain product, in which the goal is to parenthesize the sequence of matrices so as to minimize the number of scalar multiplications. (b) Illustrate your algorithm by the sequence of dimensions {5, 10, 3, 12, 5, 50, 6}. (c) Consider a variant of the matrix-chain multiplication problem in which the goal is to parenthesize the sequence of matrices so as to maximize, rather than minimize, the number of scalar multiplications. Does this problem exhibit optimal substructure?

5-7題請書寫於B答案卷

5. 24% Assume the basic knowledge about NP-completeness in the textbook.(For example; Satisfiability, Clique, Vertex-cover, Hamiltonian-cycle, and Subset-sum problems are NP-complete.) For each of the following problems, answer whether it is in P, NP-complete, or NP-hard. If it is in P, give the name of the algorithm or write a few words if necessary (no need to prove it). If it is NP-hard, you MUST give a proof.

(1.)Find a longest simple path between two nodes, where all the edge weights are positive.
(2.)Find a shortest simple path between two nodes, where the edge weight can be negative or positive.
(3.)Find a negative-weight cycle.
(4.)Find a negative-weight cycle with the most negative total weight.
(5.)Find a largest cycle in a graph, where the edge weight is 1 for every edge.
(6.)Find a smallest cycle in a graph, where the edge weight is 1 for every edge

6. Consider the minimum spanning tree problem.

(1.)6% Prove the correctness of the Kruskal's algorithm.

(2) 2% There is a special data structure to implement the Kruskal's algorithm more efficiently. Give the name of this data structure, and give the time complexity of each operation used in this data structure. (Just give the answer, no need to prove it.)

(3.) 6% Given a graph G and a minimum spanning tree T, suppose that we decrease the weight of one of the edges in T. Show that T is still a minimum spanning tree for G. More formally, let T be a minimum spanning tree for G with edge weights given by weight function w. Choose one edge $(x, y) \in T$ and a positive number k, and define the weight function w' by

$$w'(u,v)=\begin{cases} w(u,v) & \text{if } (u,v)\neq(x,y) \\ w(x,y)-k & \text{if } (u,v)=(x,y) \end{cases}$$

Show that T is a minimum spanning tree for G with edge weights given by w'.

7. 12% Amortized analysis

Consider the dynamic table problem (with both insertion and deletion operations) introduced in the textbook. (Review the problem briefly as follows: When the table is full, expand the table to a new table with double size. When the load factor is decreased down to $\frac{1}{4}$, $\alpha(T) < \frac{1}{4}$, contract the table down to a new table with half size.)

For each of the following possible potential functions, indicate the worst case that causes the largest amortized cost $\hat{c_i}$ for one operation, and compute this $\hat{c_i}$.

$$\phi(T) = \begin{cases} s(num[T] - \frac{1}{2}size[T]) & \text{if } \alpha(\mathrm{T}) \geq \frac{1}{2} \\ t(\frac{1}{2}size[T] - num[T]) & \text{if } \alpha(\mathrm{T}) < \frac{1}{2} \end{cases}$$

(1.) $s = 2$，$t = 1$

(2.) $s = 1$，$t = 1$

(3.) $s = 3$，$t = 1$