

科目：演算法 A

日期：103 年 1 月 23 日 第 1 頁 共 1 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. Asymptotic notations:

- Given two sorted lists of length $n/2$, which one of $O(n)$ and $\Theta(n)$ is the best to describe the time required for merging two lists into one sorted list? 4%
- To insertion sort a list of length n , which one of $O(n^2)$ and $\Theta(n^2)$ is the best to describe the computing time. 4%
- What is the result of $O(n^2) + \Theta(n^2)$? 4%

2. Lower bound:

- How do we establish the $\Omega(n \log n)$ lower bound for any sorting algorithm that sorts using comparison operation? 8%
- Two problems A and B , Problem A is to compute the nearest neighbours among a set of n numbers S ; Problem B is to identify if there are two numbers in the set S are identical. My question is, we can establish the lower bound to A by using the known lower bound to B or we can establish the lower bound to B by using the known lower bound to A . 6%
- Show that Radix sort correctly sorts n numbers of k digits (k is a constant) in linear time. You have to argue that it takes linear time and it correctly sorts. 10%
- In question a., we established the lower bound. In question c., we had a linear time algorithm for sorting. Anything went wrong? 6%

3. Build a min-heap:

- Describe the algorithm to build a min heap in linear time. 8%
- Show the linear time bound. 10%

◎請用深黑色鋼筆或原子筆出題

命題老師簽名：

科目：演算法 B

日期：103 年 1 月 23 日 第 1 頁 共 1 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% Use dynamic programming technique to find the longest monotonically increasing subsequence of a sequence of n numbers.
(Write the object function, recursive relation, initial condition, and the answer. Give the time and space complexity of your algorithm.)
2. 16% Let $G(V, E)$ be an undirected finite graph, where each edge e has a given length $l(e) > 0$. Let s be a fixed vertex of this graph and let $\delta(v)$ be the length of a shortest path from s to v .
 - (1) Prove that for each vertex $v \neq s$, there is an incident edge $e = (u, v)$ such that $\delta(v) = \delta(u) + l(e)$.
 - (2) Show that if such an edge, as in part (1), is chosen for each vertex $v \neq s$ then the set of edges forms a spanning tree of graph G .
 - (3) Prove or disprove that the spanning tree obtained in part (2) is a minimum weight spanning tree.
3. 18% For each of the following three maximization problems, decide whether it is a P-problem or an NP-hard problem. If it is P, describe a polynomial time algorithm to solve it and explain your answer briefly. If it is NP-hard, explain it.
Let $G(V, E)$ be an undirected finite graph, where each edge e has a given length $l(e) > 0$ (or called weight.).
 - (1) The maximum weight spanning tree problem. (Instead of finding a minimum weight spanning tree in a weighted graph, the problem is to find a maximum weight spanning tree.)
 - (2) The longest traveling salesman tour problem. (Instead of finding a shortest traveling salesman tour in a weighted graph, we want to maximize the length of the entire trip.)
 - (3) The longest simple path problem. (Instead of finding a shortest simple path between two vertices, we want to find a longest simple path between two vertices.)

◎請用深黑色鋼筆或原子筆出題

命題老師簽名：