

※ 考生請注意：本試題不可使用計算機。 請於答案卷(卡)作答，於本試題紙上作答者，不予計分。

一、Data Structures (50%)

1. (10%) Given inorder and preorder traversal of a tree as follows.

inorder: JHKFIDGBEAC

preorder: ABDFHJKIGEC

(5%) reconstruct the binary tree

(5%) show the postorder traversal of this tree

2. (10%) Suppose there are 11 slots available from 250, please give the hashing function to insert the keys, 42, 77, 85, 113, 315, 433, 474, 479, 574, 582, 698, to the 11 slots without collision by using the division method, and show the hashing table.
3. (10%) What are the minimum and maximum numbers of elements in a heap of height h ?
4. (10%) Given an $O(n \lg k)$ -time algorithm to merge k sorted lists into one sorted list, where n is the total number of elements in all the input lists.
5. (10%) Given a sequence $K = \langle k_1, k_2, \dots, k_6 \rangle$ of 6 distinct keys in sorted order with probabilities 0.06, 0.08, 0.10, 0.04, 0.12, 0.14. Some searches may be for values not in K , and so we also have 7 dummy keys, d_0, d_1, \dots, d_6 , with probabilities 0.07, 0.07, 0.07, 0.06, 0.06, 0.06. Because we have probabilities of searches for each key and each dummy key, we can determine the expected cost a search in a given binary search tree T . Suppose the expected cost of a search in T is

$$E[\text{search cost in } T] = \sum_{i=1}^n (\text{depth}_T(k_i) + 1) \cdot p_i + \sum_{i=0}^6 (\text{depth}_T(d_i) + 1) \cdot q_i, \text{ where}$$

depth_T denotes a node's depth in the tree T . Determine the cost of an optimal binary search tree.

※ 考生請注意：本試題不可使用計算機。請於答案卷(卡)作答，於本試題紙上作答者，不予計分。

二、Algorithms (50%)

6. (18%) Answer each part TRUE or FALSE for the little o notation.
- a) (3%) $n = o(8n)$.
 - b) (3%) $2^n = o(n^3)$.
 - c) (3%) $2^n = o(4^n)$.
 - d) (3%) $1 = o(n^2)$.
 - e) (3%) $n = o(\lg n)$.
 - f) (3%) $1 = o(\frac{1}{n})$.
7. (12%) Given an $O(V)$ -time algorithm that determines whether or not a given undirected graph $G=(V,E)$ contains a cycle.
8. (10%) Use the master method to give tight asymptotic bounds for the following recursive $T(n) = 7T(\frac{n}{2}) + \Theta(n^2)$.
9. (10%) A directed graph $G=(V,E)$ is *semiconnected* if, for all pairs of vertices $u, v \in V$, u is reachable from v through a directed path, or v is reachable from u through a directed path. Given a linear-time algorithm to determine whether or not G is semiconnected.