總分100分

1. Binary search trees find their widely use in many applications. Answer the following short questions about binary search tress. (20%)

   (a) Define a binary search tree. (5%)

   (b) Using data sequence: {40, 20, 60, 10, 30, 50, 70, 45, 55, 52}, construct a binary search tree. (5%)

   (c) Write a recursive search function for binary search trees, assuming that the input arguments are the root pointer and an integer key. The return value is the pointer of the node containing the key or null. (5%)

   (d) Explain how to use binary search trees to sort sequences. What is the time complexity of it? (5%)

2. Queues and priority queues are two widely used data structures. Answer the following short questions. (20%)

   (a) Describe the differences between queues and priority queues. (5%)

   (b) Define a min-priority queue. (5%)

   (c) What are the time complexities of insertion and deletion operations when a min-prority queue is represented in terms of the following data structures: an unordered array, an unordered linked list, a sorted array, a sorted linked list, and a min heap? (10%)

3. Answer the following two short questions about stack and min-priority queue. (10%)

   (a) The stack is also a data structure. Describe its main features. (5%)

   (b) Can a min-priority queue be used as a stack? If yes, please explain how to do this. (5%)

國立台灣科技大學九十七學年度碩士班招生試題

系所組別： 電子工程系碩士班甲組

科　　目： 資料結構

4. (20%)

(a) Briefly explain the hashing technique and discuss its advantages. (5%)

(b) Discuss the advantages and disadvantages of the two clash (collision) resolution strategies: open addressing and chaining. (5%)

(c) What are the key points for designing a good hash function? (5%)

(d) What would be a good hash function for license plates that is a string of numbers and letters of the form " ＊＊○○○＊＊", where an "○" represents a digit and an "＊" represents a letter? Why? (5%)

5 (20%)

(a) Write an algorithm that checks if a complete binary tree is a heap. Analyze your algorithm and show the result using order notation. (10%)

(b) Design a sorting algorithm based on a ternary heap. A ternary heap is like an ordinary heap except that each internal node has three children. (10%)

6. (10%)

Write a C/C++ function that takes an integer array and its size as arguments and finds the minimum and the maximum of the numbers. This can be done by using no more than $3n/2$ comparisons where n is the size of the array.