國立交通大學試題紙

九十六學年度第一次
博士班資格考

科目：演算法（A）　　日期：97 年 1 月 25 日　第 1 頁 共 2 頁

1. Consider the minimum spanning tree problem
   (a). Prove the correctness of the Kruskal's algorithm. (It is also called a greedy algorithm for the minimum spanning tree problem.) 8%
   (b) In order to implement the Kruskal's algorithm efficiently, a special kind of data structure is used, write the name of this data structure. 1%
   (c) Using this data structure, write the time complexity of each Union operation. Explain briefly the basic idea to union two disjoint sets. What advantage does it have? 3%
   (d) Using this data structure, write the worst case time complexity of each Find operation. Explain briefly the basic idea to achieve this time complexity. 3%

2. Consider the shortest path problem. (In a directed graph, some of the edges may have negative weights.) Assume $P \neq NP$.
   (a) Describe briefly the Bellman-Ford algorithm. Prove the correctness of this algorithm. State and explain briefly the time complexity.
   Can this algorithm always find a shortest simple path between two nodes?
   No matter what the edge weights are? Explain or prove it briefly. 9%
   (b)Is the following problem a P-problem or an NP-hard problem? Explain briefly.
   "Find a negative weight directed cycle." 4%
   (c)Is the following problem a P-problem or an NP-hard problem? Explain briefly.
   "Find a directed cycle with the most negative total weight." 4%

3. (a)Which of the following 12 functions are polynomially bounded?
   (b)Which are polylogarithmically bounded?
   6%　每錯一個扣一分

   1. $(\log n)!$　2. $\log(n!)$　3. $(\log n)^{\log n}$　4. $(n)^{\log\log n}$　5. $2^{\sqrt{2\log n}}$

   6. $4^{\log n}$　7. $n^{1/\log n}$　8. $2^{\log^* n}$　9. $\log^*(\log n)$　10. 1

   11. $1+\frac{1}{2}+\frac{1}{3}+\frac{1}{4}+\ldots+\frac{1}{n}$

   12. $n^{(1+\frac{1}{2}+\frac{1}{3}+\frac{1}{4}+\ldots+\frac{1}{n})}$

國立交通大學試題紙

九十六學年度第一次
博士班資格考

科目：演算法 (A) 日期：97 年 1 月 25 日

4. Give asymptotic upper bounds for $T(n)$ in each of the following recurrences. Assume that $T(n)$ is constant for $n \le 2$. Make your bounds as tight as possible.
不能只寫答案，必須要有一些簡要的推導，或引用基本知識以顯示你的答案是有根據推論出來的。(例如 Master Theorem 就可當作基本知識直接引用，不必證明．)

(a) $T(n) \le \frac{1}{2}\sqrt{n}\,T(4\sqrt{n}) + cn\log n$ 6%

(b) $T(n) = 2\sqrt{n}\,T(\sqrt{n}) + c\dfrac{n\log n}{\log\log n}$ 6%

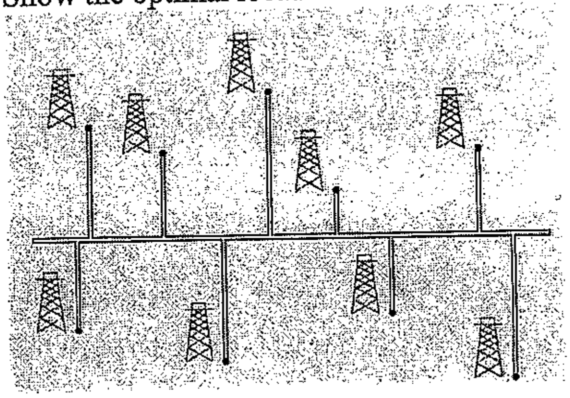
附註：(a)這個 recurrence 是在研究 Fast Fourier Transform 過程中，所推導出而面臨到的 recurrence。

國立交通大學試題紙

九十六學年度第一次
博士班資格考

科目：演算法（B）　　日期：97 年 1 月 25 日　第 1 頁 共 1 頁

1. (8%) An oil company is planning a large pipeline running east to west horizontally through an oil field of n wells. From each well, a spur pipeline is to be connected directly to the main pipeline along a shortest path (either north or south), as shown below. Given coordinates (x, y) of the wells, how should the company pick the optimal location of the main pipeline (the one that optimizes the total length of the spurs)? Show the optimal location can be determined in linear time.



2. (10%) Now, you are inside a Buffet restaurant. Assume that your stomach can only accept food with maximum size M and there are n kinds of food with sizes and values as (s_1, v_1), (s_2, v_2), …, (s_n, v_n), in the restaurant. The restaurant requests all customers to eat up all foods taken without splitting. Otherwise, get some high penalty. Assume all values are integers. Design an algorithm to find the highest value of food that you can take and eat. Demonstrate your algorithm based on the case: M=27 and 4 kinds of food with (3, 4), (4, 5), (6, 9), (8, 13). Hint: this is equivalent to the knapsack problem.
3. (20%) Design data structures, called Hashtable, which require the following operations (no deletion for simplicity)

   (1) `find(key)`: returns the data with `key`;

   (2) `insert(key, data)`: insert `data` with `key` into the hashtable.

   Assume that each key value is less than a sufficiently large prime number p. Design the data structure, Hashtable, such that

   (a) the averaged time complexity for each `find` operation is O(1),

   (b) the amortized time complexity for each `insert` operation is O(1) by assuming that each `find` operation takes O(1) time, and

   (c) the space complexity is always O(n) if there are n characters in the string buffer.

   And explain why (a), (b), and (c) are satisfied in your data structure.
4. (12%) Assume that the 3-CNF-SAT problem has already been proved to be NP-complete. Now, prove that the clique problem is NP-complete.