國立臺北科技大學 102 學年度碩士班招生考試

系所組別：2300 資訊工程系碩士班

第二節　資料結構與演算法　試題

注意事項：

1. 本試題共九題，配分共 100 分。
2. 請標明大題、子題編號作答，不必抄題。
3. 全部答案均須在答案卷之答案欄內作答，否則不予計分。

一、(10 pts) In all of the five recurrences shown below, it is assumed that $T(1) = d$ for some constant d. State, using the "big oh" notation, the solution to each of the five recurrences shown below. Just state the answer - you do NOT need to justify them.

1. (2pts) $T(n) = 5T\left(\frac{n}{3}\right) + 4n^2$
2. (2pts) $T(n) = 16T\left(\frac{n}{2}\right) + 5n^4$
3. (2pts) $T(n) = 4T\left(\frac{n}{2}\right) + n^2 \log n$
4. (2pts) $T(n) = \sqrt{n}T(\sqrt{n}) + n$
5. (2pts) $T(n) = 35T(\frac{n}{3}) + 7n^3$

二、(10 pts) Please uses substitution method to Show that the solution to

$$T(n) = 2T\left(\left\lfloor\frac{n}{2}\right\rfloor + 17\right) + n$$

is $O(n \log n)$.

三、(10 pts) Please give the best asymptotic running time for each of the problems using the "big Oh" notation. If you think the problem is NP-complete, state so (no running time should be given in this case). Just state the answers- you do NOT need to justify them.

1. (2pts) Finding the median in an unsorted set of size n.
2. (2pts) The *union* operation on a Fibonacci heap with n elements using amortized analysis.
3. (2pts) Given a boolean formula in conjunctive normal form (*i.e.*, $C_1 \wedge C_2 \wedge \ldots \wedge C_k$, where every C_i contains an arbitrary number of literals $\vee$-ed together), determine whether there exists a truth assignment to the variables satisfying the formula.
4. (2pts) Perform a Breadth-first search on a graph $G = (V, E)$, where $|V| = n$ and $|E| = m$.
5. (2pts) Sorting n d-digit numbers in which each digit can take on up to k possible values using Radix-Sort.

四、(10 pts) Mark by T(=true) or F(=false) each of the following:

1. (2pts) If algorithm A solves problem L in $O(n \log n)$ time, then no instance of problem L can, when given to A as input, makes A take n^2 time.
2. (2pts) Suppose problem P_1 can be reduced to problem P_2 in linear time (*i.e.*, $P_1 \propto_{O(n)} P_2$). Then, if there exists a polynomial time algorithm for P_1, then there exists a polynomial time algorithm for P_2.
3. (2pts) If an NP-complete problem can be solved deterministically in $O(n^3)$, then every problem in class NP can be solved in $O(n^3)$.
4. (2pts) If any NP-complete problem is not in P, then no NP-complete problem is in P.
5. (2pts) Suppose that "if we could solve problem A in time $O(T(n))$, then we could solve problem B in time $O(n \log n+T(n))$". If A has an (n^2) time lower bound then B does too.

五、(15 pts) Please answer each of the following problems shortly and concisely.

1. (5 pts) Give an n-element array A of real numbers, design an $O(n)$ time algorithm which determine whether any value occurs more than $\frac{n}{7}$ times in A.
2. (5 pts) Recall the disjoint set data structure. Suppose now we are given a graph. Show how to use the disjoint set with its operations to compute the connected components of the input graph. Please give the pseudo-code and the time complexity of your approach.
3. (5 pts) Let T be an n-node tree that is binary and has height h. For each node v in T, let $SIZE(v)$ be the number of nodes in the subtree rooted at v in T (including node v). Assume that the $SIZE$ array is already given to you. Give an $O(h)$ time algorithm for finding a node v in T such that $\frac{n}{3} \leq SIZE(v) \leq \frac{2n}{3}$.

注意：背面尚有試題

六、(10 pts) Suppose that we use bottom-up (heapfy) heap construction to build up a minheap with an array shown as below.

14	9	8	25	5	11	27	16	15	4	12	6	7	23	20

1. (5 pts) Please show the construction step by step (i.e. draw the heap construction).
2. (5 pts) On the constructed heap, please show how to perform the operation removeMin by drawing.

七、(10 pts) Consider the DAG in Figure 1. Suppose we use the DFS-based approach to find the topological sort starting with node B. Since we use DFS on the dag, we can have a DFS-tree T rooted at B. Let $d[u]$ denote the discovery time and $f[u]$ denote the finishing time for each node u in the dag during the execution of DFS. Please answer the following questions.

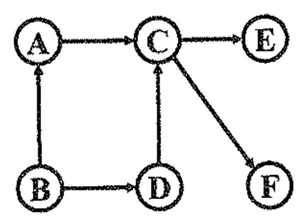

Figure 1. A DAG (Directed Acyclic Graph)

1. (4 pts) How many different topological sorts starting from node B?
2. (3 pts) The interval $[d(E), f(E)]$ is either in the interval $[d(A), f(A)]$ or disjoint with the interval $[d(A), f(A)]$ for all the possible DFS's. True or False?
3. (3 pts) How many cross edges with respect to T?

八、(10 pts) Given integers n and k, along with $p_1, p_2, \ldots, p_n \in [0, 1]$, you want to determine the probability of obtaining exactly k heads when n biased coins are tossed independently at random, where p_i is the probability that the ith coin comes up heads. Give an $O(nk)$ algorithm for this task. Assume you can multiply and add two numbers in [0, 1] in O(1) time.

九、(15 pts) Consider the weighted, directed graph in Figure 2 and the single-source shortest-paths problem. Please answer the following questions when applying the Bellman-Ford algorithm:

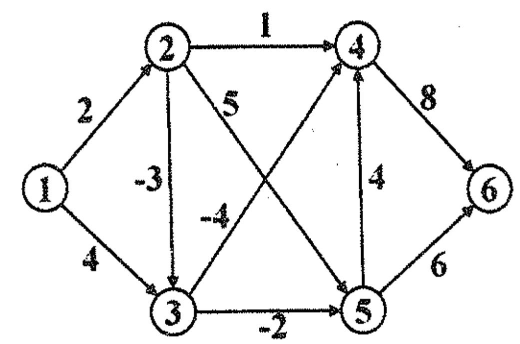

Figure 2. A weighted, directed graph

1. (3 pts) Please show the result when using the Bellman-Ford algorithm to find the shortest paths from node 1 to every other node step by step.
2. (3 pts) What is the running time of the Bellman-Ford algorithm on a graph $G=(V, E)$ where $|V|=n$ and $|E|=m$?
3. (3 pts) Suppose the result of the ith iteration is the same as the result of the $(i+1)$th iteration where $i \leq |V|-1$. Please justify whether the algorithm can stop at this time point and we can conclude the result is correct. If it is true, please show it; otherwise, give a counter-example.
4. (6 pts) Suppose that a weighted, directed graph $G=(V, E)$ has a negative-weight cycle. Give an efficient algorithm to list the vertices of one such cycle. Prove that your algorithm is correct.