國立臺北科技大學 101 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　離散數學與演算法　試題

注意事項：

1. 本試題共十題，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。

1. (10 %) Solve the following difference (recurrence) equation:

$$x_n + x_{n-1} + \frac{1}{4}x_{n-2} = (\frac{1}{4})^n, n \geq 0, \text{ with } x_0 = 1 \text{ and } x_1 = 1.$$

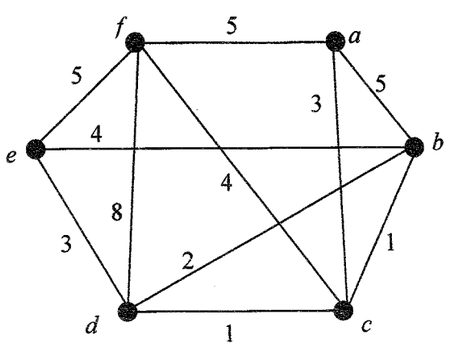
2. (10%) The lattice points in plane $\mathbf{R}^2$ are the points (x, y) where x and y are both integers. Prove that there are only countably many lattice points in $\mathbf{R}^2$.

3. (15%) Let R be a binary relation on a set X. The n-th power of R, denoted R^n, is defined as follows:

$$R^0 = \{(x, x) : x \in X\}$$
$$R^n = R \circ R^{n-1} = \{(x,z) : (x,y) \in R^{n-1} \text{ and } (y,z) \in R\}$$

Show that $\bigcup_{i=1}^{\infty} R^i = R^+$ is the smallest transitive relation containing R.

4. (5%) Draw a minimum cost spanning tree for the graph G shown below and then find its minimum cost.



5. (10%) How many integers between 1 and 250, including 1 and 250, are divisible neither by 3 nor by 7 but are divisible by 5?

6. (10%) Please give the best asymptotic running time for each of the problems or bound for each of the recurrences below, using the "big Oh" notation. It is assumed that $T(1)=d$ for some constant d in all the recurrences. Just state the answers- you do not need to justify them. (2% for each)
   (1) $T(n)=10T(n/3)+5\ n^2$
   (2) $T(n)=2T(n/4)+\ n^{(1/2)}$
   (3) Multiply two $n \times n$ matrices.
   (4) In a directed, weighted graph $G = (V, E)$ with positive weights and $|V| = n$ and $|E| = m$, determine the shortest path between a given pair of vertices.
   (5) Given a boolean formula in disjunctive normal form (*i.e.*, $C_1 \vee C_2 \vee \ldots \vee C_k$, where every C_i contains an arbitrary number of literals $\wedge$-ed together), determine whether there exists a truth assignment to the variables satisfying the formula.

7. (10%) Mark by T(=true) or F(=false) each of the following : (2% for each)
   (1) If problem P has an $\Omega(n^2)$ lower bound then, for any algorithm A that solves P, there exists one instance of problem P that, when given as input to A, makes A takes $\Omega(n^2)$ time.
   (2) If a problem that is in the class NP has a polynomial time solution, then P is equal to NP.
   (3) If someone was able to give an exponential time lower bound for a problem that is NP-complete, then this would imply that P is not equal to NP.
   (4) If a problem is NP-complete, this implies that such a problem is no solution at all.
   (5) Suppose problem P_1 can be reduced to problem P_2 in linear time. Then, if P_1 is NP-hard then P_2 is NP-hard.

8. (10%) Please answer each of the following problems shortly and concisely.
   (1) (5%) Solve the continuous-knapsack problem for the following weights(w_i), profits(p_i), and knapsack capacity C. Please show your work **step by step**.

i	1	2	3	4
w_i	12	15	20	15
p_i	4	3	6	8

C=50

   (2) (5%) An ***inversion*** in an array A is a pair of indices (i, j), $i<j$, with $A[i]>A[j]$ (*i.e.*, a pair of indices that references values that are out of order). Given an array of size n, we would like to count the inversions. Please provide an O(n log n)-time algorithm to count the number of inversions.

注意：背面尚有試題

9. (10%) Please answer each of the following problems shortly and concisely.
   (1) (3%) Are there graphs for which Prim's algorithm is faster than Kruskal's algorithm?
   (2) (3%) Does Dijkstra's algorithm allow the edges having negative weight in the input graph? Please give the reasons to support your conclusion.
   (3) (4%) Give an algorithm that determines whether or not a given undirected graph $G=(V, E)$ contains a cycle. Your algorithm should run in $O(|V|)$ time, independent of $|E|$.

10. (10%) Let T be an n-node tree rooted at some node r. We want to place as few *guards* as possible on nodes in T, such that every edge of T is *guarded*: an edge between a parent node v and its child w is *guarded* if one places a guard on at least one of these two nodes v, w. Give an $O(n)$ time algorithm for finding an optimal solution to the problem. Please show the analysis on the time and correctness of your algorithm.