類別：資工類　　科目：資料結構與演算法　　共 2 頁 第 1 頁　　＊請在試卷答案卷（卡）內作答

本科考試禁用計算器

1. A is a 2-dimensional array of real numbers with 100 rows and 100 columns. Assume that each real number occupies 4 bytes, and the address of A[0, 0] is $B000_{16}$. The index of both row and column is assumed to start from 0.
   (1) Compute the address of A[43, 21] if the array A is stored row-major. (5 points)
   (2) Compute the address of A[43, 21] if the array A is stored column-major. (5 points)

2. Give 2 different binary trees, each with three nodes, that have the same preorder and postorder sequences. (10 points)

3. Given 6 external nodes A, B, C, D, E, and F, and their corresponding weights 2, 3, 5, 7, 9, 13, respectively.
   (1) Use Huffman's algorithm to construct a binary tree with the minimal total weighted external path length. (10 points)
   (2) Compute the total weighted external path length of the constructed tree in part (1). (5 points)

4. Provide three different data structures to represent the graph in Figure 1 with 4 nodes. (15 points)

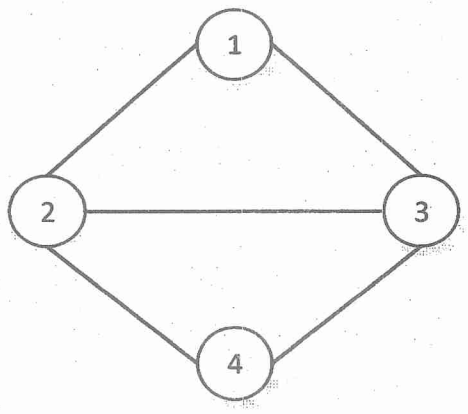

Figure 1

5. A *tree* is a connected undirected graph $G = (V, E)$ which contains no cycles. Many problems can be solved efficiently in trees. For example, it takes super-linear time to solve the maximum matching (a *matching* is a set of edges without common vertices) problem in general graphs or bipartite graphs. However, in trees, the problem can be solved in linear time. In the input file, each tree is given in several lines. The first line contains an integer n indicating the number of vertices

參考用

注意：背面有試題

(denoted by numbers 1, 2, …, n) in the tree. The following $n-1$ lines contain the edges given by a pair of vertex numbers (recall that each tree with n vertices has exactly $n-1$ edges), which are separated by a space.

(1) "Do not assume the input is always valid." It is a good advice for programmers. For example, consider the following input:

5
1 2
2 3
1 3
4 5

The input indicates that there is a graph with 5 vertices and 4 edges. Note that this graph is not a tree because it is not connected and has a cycle. Design an $O(n)$ time algorithm to detect whether the given input represents a tree. (7 points)

(2) Argue that the following observation is correct: For each leaf v and the only edge (u,v) incident to it, there exists a maximum matching $M \subseteq E$ such that $(u,v) \in M$. (6 points)

(3) Based on the observation in (2), give an $O(n)$ time algorithm to solve the maximum matching problem in trees. Please describe what data structure is used for storing the given tree, and what mechanisms are used in its implementation to achieve the linear running time. (12 points)

6. Given a set of n planar points and the corresponding Voronoi diagram D, write an O(n)-time algorithm to find out the closest pair of points. Note that you can use the fact that the closest pair of points must share a Voronoi edge in the Voronoi diagram and there are at most 3n-6 Voronoi edges. (12 points)

7. The sum of subsets problem is described as follows. Given a set S of n integers and a specific integer M, is there a subset of the integers that sum to M? Write a non-deterministic algorithm to solve the sum of subsets problem in O(n) time. (13 points)