國立成功大學 107 學年度碩士班招生考試試題

系　　所：資訊管理研究所

考試科目：資料結構

考試日期：0206，節次：3

※ 考生請注意：本試題不可使用計算機。　請於答案卷(卡)作答，於本試題紙上作答者，不予計分。

A1. [18%] A priority queue is a data structure that supports storing a set of values, each of which has an associated key. Each key-value pair is an entry in the priority queue. The basic operations on a priority queue are:

- insert(k, v) – insert value v with key k into the priority queue
- removeMin() – return and remove from the priority queue the entry with the smallest key

Other operations on the priority queue include *size*(), which returns the number of entries in the queue and *isEmpty*() which returns *true* if the queue is empty and *false* otherwise. Two simple implementations of a priority queue are an unsorted list, where new entries are added at the end of the list, and a sorted list, where entries in the list are sorted by their key values. Fill in the following table to give the running times of the priority queue operations for these two implementations using big O notation. You should assume that the implementation is reasonably well done.

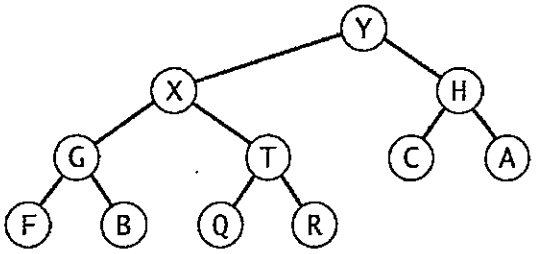
Operation	Unsorted List	Sorted List
size, isEmpty		
insert		
removeMin		

A2. [12%] The nodes in an integer binary tree can be represented by the following data structure.

```
public class Node {      // Binary tree node
    public int value;    // data in this node
    public Node left;    // left subtree or null if none
    public Node right;   // right subtree or null if none
}
```

Implement a function to perform breath-first traversal of a binary tree and print the data value of each node in the order they reached during the traversal. You may create and use instances of other standard data structures (e.g., linked lists, hash tables, or whatever) and their operations if they are useful, without having to give details of their implementations.

A3. [10%] Consider the following binary tree representation of a max-heap.



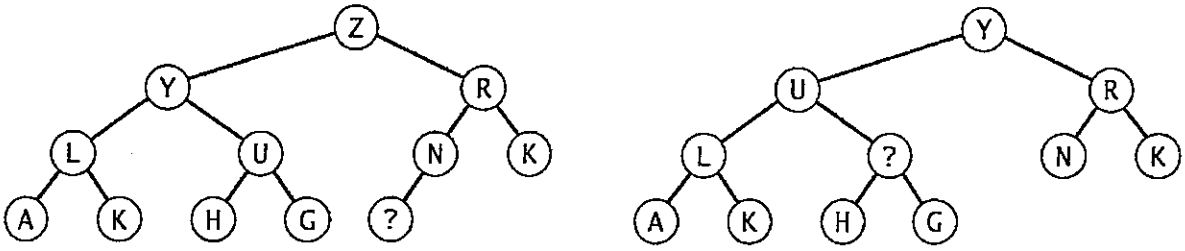
(a) [3%] Give the array representation of the heap

0	1	2	3	4	5	6	7	8	9	10	11	12
-												-

(b) [3%] Insert the key *P* into the binary heap above, circling any entries that changed

0	1	2	3	4	5	6	7	8	9	10	11	12
-												-

(c) [4%] A *delete-the-max* operation in the binary heap at left results in the binary heap at right.



What could be the key at the node with a question mark? Give all possibilities.

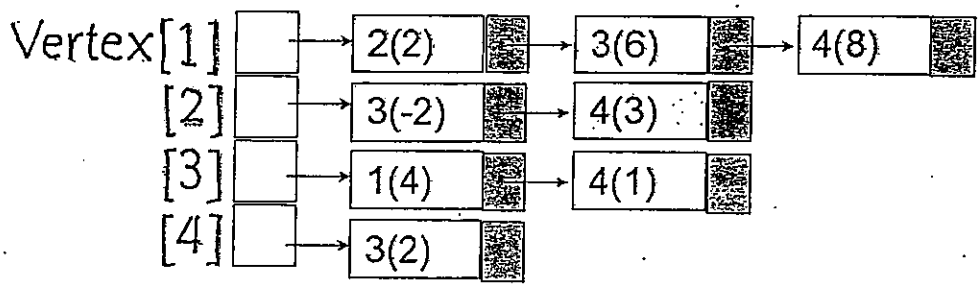
A4. [10%] A square matrix is called upper triangular if all the entries below the main diagonal are zero. Design a data structure in C++ that represents upper triangular matrices. Your matrix data type should allow efficient access to matrix entries and support addition and multiplication operations. You may assume that matrix entries are all integers.

※ 考生請注意：本試題不可使用計算機。　請於答案卷(卡)作答，於本試題紙上作答者，不予計分。

B1. A graph adjacency matrix is shown in the figure below. Please use Sollin's method to find out the spanning tree (Please show the process in detail) (8%). When constructing an n-vertex spanning tree, you need to avoid cycle path. How to choose an edge (u, v) and avoid this edge causing a cycle? (4%)

	1	2	3	4	5	6	7	8
1	0	2	8	-	-	-	-	-
2	2	0	7	9	-	-	-	-
3	8	7	0	4	10	12	-	-
4	-	9	4	0	-	-	-	-
5	-	-	10	-	0	6	14	-
6	-	-	12	-	6	0	-	-
7	-	-	-	-	14	-	0	3
8	-	-	-	-	-	-	3	0

B2. The following figure is the adjacency list of a graph. In this figure, every chain node shows the vertex number and its distance by **vertex number(distance)** (i.e. first node 2(2) means vertex 1 connect to vertex 2 with distance 2) in the adjacency list. How to find all the pairs shortest path by Floyd's algorithm? Please show the process in details. (10%)



B3. Big data is a hot issue now. When the quantity of data to be sorted is larger than the memory we have, external sort will be considered. Assume the computer memory size is 10 blocks and your data is 200 blocks. Please explain how many runs will be generated in the Pass 0 (run generation) (3%). In pass 0, runs are generated by internal sort. Assume there is a data set = {3, 2, 5, 7, 9, 4, 1, 10, 8}. Please explain how to complete the sorting by quick sort with in-place memory partitioning (Please discuss the detail process of quick sort) (8%) After run generation, please discuss how many merging passes are needed for 2-way (3%) and 9-way (3%) merging respectively.

B4. In hash table, when overflow occurs, linear probing could be a way to solve. Assume the hash function is key % 13. Please explain how to deal with the following data {6, 12, 25, 18, 7, 30, 14, 1, 13} in a one dimension array a[0..12] table. (6%) Please draw the result when 25 is deleted from this linear probing hash table. (5%)