國立中山大學101學年度碩士暨碩士專班招生考試試題

科目：資料結構【資訊管理學系碩士班】

題號：4119

1. (10%) True/False questions. Please justify your answer. No justification, no points.
   (A) Every binary tree is uniquely defined by its pre-order and post-order sequences. (5%)
   (B) For a given graph and a given starting node, there is a unique minimum spanning tree. (5%)

2. (15%)
   (A) What are the time complexities of AVL search tree for finding a key in the best and worst cases? (8%)
   (B) What are the time complexities of hash table for finding a key in the best and worst cases? (7%)

3. (15%) Suppose you need a data structure, either an AVL search tree or a hash table with separate chaining to represent a string, to support several types of operations on a set of input: insert a key, find a key, and print all the key values in order.
   (A) Suppose the operating environment has high insertion rate, high search rate, and high printing rate. Which data structure will you choose? Why? (8%)
   (B) Suppose the operating environment has low insertion rate, high search rate, and very low printing rate. Which data structure will you choose? Why? (7%)

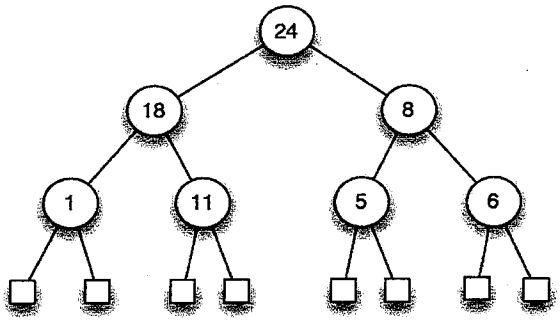
4. (10%) Below is a partial code for priority queue using heap. Please write down the missing code in the blank. Explain your code.

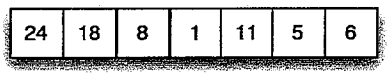
```
/**
* Priority Queue Structure
*/
typedef struct PQueue_s {
   size_t size;          /* The actual size of heap at a certain time */
   size_t capacity;      /* The amount of allocated memory for the heap */
   int  data;        /* data stored in max-heap */

} PQueue;

/* Util macros */
#define LEFT(x) (2 * (x) + 1)
#define RIGHT(x) (2 * (x) + 2)
#define PARENT(x) ((x) / 2)

/* Adds a new element to the Priority Queue. */
void pqueue_enqueue(PQueue *q, int data) {
   size_t i;
   int tmp;
   if (q->size >= q->capacity) {
      DEBUG("Priority Queue is full. Cannot add another element .");
```

```
        return;
    }

    /* Adds element last */
    q->data[q->size] =  data;
    i = q->size;
    q->size++;

    /* The new element is swapped with its parent as long as its
    precedence is higher */
```

```
    }
}
```

5. (10%) Consider the following Heap, together with its array representation:





Suppose we perform the following sequence of operations on the heap: removeMax(), insertItem(17), removeMax(), insertItem(5), insertItem(9). What is the order of the elements in the heap array after this series of operations have been performed?

6. (10%) Write a procedure that uses a one-dimensional array A to store the following matrix.

$$\begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{bmatrix}$$

7. (15%) An undirected graph $G = (V, E)$ is said to be bipartite if it is possible to partition V into two sets V_1, V_2 ($V = V_1 \cup V_2$, $V_1 \cap V_2 = \{\ \}$) such that for every edge (u, v) in E, one of $\{u, v\}$ belongs to $V1$, and the other belongs to V_2. Describe an algorithm which, given a

graph $G = (V, E)$, runs in $O(n+m)$ and determines whether G is a bipartite graph or not.

8. (15%) Consider a binary search tree where each node contains the following attributes of a course: course code, course name and course grade average. The binary search tree is ordered according to the course grade average. Define an appropriate node structure for the binary search tree, and then write a procedure which swaps the content of the root node and the content of the node with the highest course grade average.