國立台灣海洋大學九十學年度研究所碩士班招生考試試題

系所名稱：資科系碩士班　　＊答案以橫式由左至右書寫於答案卷上！

科目名稱：基礎計算機科學

Part 1: 離散數學（共 4 題，題號 1-4, 共計 40 分）

Part 1: Discrete Math. (There are 4 problems, and the total points are 40 points.)

1. | A | =6, | B | =6. （10 points）
   (a) How many functions are there from A to B?
   (b) How many 1-1 functions are there from A to B?
   (c) How many onto functions are there from A to B?

2.(a)State and prove Chinese Remainder Theorem. （10 points）
   (b)Find the smallest number with
   $X \equiv 1 (\mathrm{mod}\, 3)$
   $X \equiv 4 (\mathrm{mod}\, 5)$
   $X \equiv 2 (\mathrm{mod}\, 7)$

3.How many non-negative integer solutions $x_1 + x_2 + x_3 \le 100$ （10 points）

4.Solve the recurrence relation $a_n = 6a_{n-1} - 8a_{n-2} + 3^n$, $a_0 = 1$, $a_1 = 2$ （10 points）

Part 2: 資料結構(共 5 題，題號 5-9, 共計 60 分)

Part 2: Data Structure (There are 5 problems, and the total points are 60 points.)

5 G = (V, E) is an undirected graph. Please design a non-recursive depth-first-search traversal algorithm in G by using a stack. However, the size of the stack, k, is much smaller than the number of vertices, |V|. (The stack can hold only a few vertices.)
(a) What kind of data structure do you use to represent G?
(b) Write down the pseudo-code of your algorithm.
(c) What is the time complexity of your algorithm? Explain the reason based on your data structures and algorithm.
(d) What kind of structure will be generated by your algorithm ? (Note that the stack size is limited.) Explain the reason. (15 points)

6. A binary tree T is *height-balanced* if its left and right sub-trees are height-balanced and |hL – hR| ≤ 1, where hL and hR are the heights of the left and the right sub-trees. Assume the data type of the tree nodes is:

```
typedef struct _node{
    int   data;
    struct _node *left, *right; // pointers to the left and right subtrees.
} node_t;
```

Design a recursive algorithm to check whether a tree is height-balanced. Write down the pseudo-code of your algorithm. The procedure name of the algorithm must be:
*Boolean check_balance(node_t *root){...}* (10 points)

7. Array Heap[] contains N random numbers. Design the following algorithms and write down their pseudo-codes:

(a) The algorithm to convert Heap[] into a max-heap structure.

(b) The algorithm to delete the maximum element from the heap.

(c) The algorithm to insert an element into the heap.(12 points)

8. Assume T is an empty red-black tree. Draw T after each datum in the following set has been inserted into it:

{X, A, D, Y, E, G, N, B, F, L}. (10 points)

9. There are 13 numbers contained in five sets: S0={2,3,9}, S1={0,10}, S2={1}, S3={5, 7, 8}, and S4={4,6,11,12}. Each set is represented by a tree structure, and the first number is served as the root. Assume procedure *union(I,J)* unions the sets rooted at I and J, and procedure *find(I)* returns the root of the tree containing *I*.

A procedure, *Super_union(I, J),* is defined as the followings:

```
Super_union(I,J){
   If(I is a root) ti = I; else ti = find(I);
   If(J is a root) tj = J; else tj = find(J);
   union(ti, tj);
}
```

The collapsing rule and the weighting rule are used for the union and find operations. Draw the trees after each *Super_union()* operation is completed:

(a) Super_union(0, 1),

(b) Super_union(10, 4),

(c) Super_union(8, 2),

(d) Super_union(3, 6).

Assume that there are N numbers , N<1,000,000 ; what is the average time complexity of each Super_union() operation ? (13 points)

以下空白