一、Data Structures (50%)

1. (20%) In this question, you have to consider the implementation of a double-ended circular queue for elements of integer type by employing an one-dimension integer array A with maximum size n. Given the following data structures:

```
int A[n];
int left;    //indicating the left boundary of the queue
int right;   //indicating the right boundary of the queue
```

During initialization, we have `right=(left+1)%n`, say, `left=0` and `right=1`.

(i) Write C-style pseudo codes for element addition and deletion operations from either end of the queue in your design.

(ii)What is the maximum queue size?

2. (20%) Consider the following C function *guess*. (Assume that *a*, *b*, and *c* are three positive integers and c is greater than one.)

```
int guess (int a, int b, int c) {
  int t1, t2, ans;
  ans=0;
  t1=b;
  while (a>0) {
    t2=a%c;
    a=a/c;
    if (t2!=0)
      ans+=t1*t2;
    t1=t1*c;
  }
  return (ans);
}
```

(i) What is the return value of the function call *guess*(11, 7, 2)? Answer the same question for the function call *guess*(11, 7, 3).

(ii) Briefly describe the purpose of the given function *guess*.

(iii)What is the time-complexity of the given function *guess*?

3. (10%) Given two sequences: ABCDEF and FBAEDC. Can you draw the unique binary tree that satisfies assumption (i)? If not, draw two distinct binary trees that can produce the sequences as (i) specifies. Answer the same questions if we use assumption (ii) instead of (i).

(i) The former is the inorder sequence of a given binary tree and the latter is the preorder sequence.

(ii)The former is the postorder sequence of a given binary tree and the latter is the preorder sequence.

(背面仍有題目,請繼續作答)

二、Algorithms (50%)

1. (18%) Answer each part TRUE or FALSE

   (a) $2n = O(n)$.

   (b) $n^2 = O(n)$.

   (c) $n^2 = O(n \log_2^2 n)$.

   (d) $n \log n = O(n^2)$.

   (e) $3^n = 2^{O(n)}$.

   (f) $2^{2^n} = O(2^{2^n})$.

2. (10%) Solving the recurrence $T(n) = 3T(\frac{n}{4}) + n\log_2 n$ using Θ notation.

3. (12%) Show that the second smallest of n elements can be found with $n + \lceil \log_2 n \rceil - 2$ comparisons in the worst case.

4. (10%) Answer TRUE or FALSE: Counting sort is a stable sort. Note that you need to explain the reason.