Assignment 8

**Date Assigned: 04/01/2019**

**Due: Midnight 04/07/2019 on iLearn**

**Please read turn-in checklist at the end of this document before you start doing exercises.**

**Section 1: Pen-and-paper Exercises**

1. Using the master theorem discussed in class, find a tight bound for the solution of the following recurrence equation (5 points each).

aT(n/b) + n^d

1. T(n) = 2T(n/2) + n3

a=2 b=2 d=3

Logb(a) = 1

d > logb(a) so O(n^d)

1. T(n) = T(9n/10) + n

a=1 b=10/9 d=1

Logb(a) = 0

d > logb(a) so O(n^d)

1. T(n) = 16T(n/4) + n2

a=16 b=4 d=2

Logb(a) = 2

d = logb(a) so O(n^d logn)

1. T(n) = 7T(n/3) + n2

a=7 b=3 d=2

Logb(a) = 1.7712437492

d > logb(a) so O(n^d)

1. T(n) = 2T(n/4) + sqrt(n)

a=2 b=4 d=1/2

Logb(a) = 0.5

d = logb(a) so O(n^d logn)

1. We are given an array of n numbers A in an arbitrary order. Design an algorithm to find the largest and second largest number in A using at most 3/2n -2 comparisons.

(i) describe the idea behind your algorithm in English (5 points);

(ii) provide pseudocode (10 points);

(iii) analyze the number of comparisons used in your algorithm (5 points).

**Note: Full credit (20 points) will be awarded for an algorithm that uses at most 3/2n -2 comparisons. Algorithms that make more comparisons will be scored out of 10 points.**

**Section 2: Java Implementation**

1. Implement problem 2 in Java (30 points).

Note:

Find a file called Problem2.java in assignment 7 folder.

Complete the method of dcfindmax2ndmax ().

Test your method in the main method provided following the comments.

**Full credit (30 points) will be awarded for an algorithm that uses at most 3/2n -2 comparisons. Algorithms that make more comparisons will be scored out of 10 points.**

**TURN-IN CHECKLIST:**

1. **Answers to Section 1 (.doc/.txt), and to Section 2 (all your source Code (.java files)). Remember to include your name, the date, and the course number in comments near the beginning of your code/report.**
2. **Create a folder and name it 'FirstName\_LastName\_assignment\_8'. In the newly created folder copy and paste your files (.doc/.txt/.java files). Then compress the folder, and push it to iLearn.**