Computer Science 431
Algorithms

The College of Saint Rose
Spring 2015

Problem Set 5: Divide and Conquer

Due: 11:59 PM, Thursday, March 18, 2015

You may work alone or in groups of size 2 or 3 on this assignment. Only one submission per group is needed.

Programming Task: Divide-and-Conquer Sorts

Extend your sorting algorithm comparison program from the previous problem sets to include options to use merge sort and quicksort. For quicksort, implement a fixed pivot (say, the first element of the range), a random pivot, and median of 3. Please note that recursive implementations will not be time or space efficient, so you should use iterative versions of these algorithms. (15 points)

Use the extended code to perform an empirical analysis of merge sort and quicksort and add these to your document describing your empirical studies. (18 points)

Written Problems

1. Levitin Exercise 5.2.5, p. 181. Justify your answers. (3 points)

2. Levitin Exercise 5.2.6, p. 181. Justify your answers. (3 points)

3. With a series of drawings (pencil and paper or a drawing program are equally acceptable), show the steps that the quickhull algorithm will use to generate the convex hull shown in Figure 5.8 on p. 196. (5 points)

4. Recall this problem from an earlier problem set: Determining the best time to buy and sell a stock is a problem of interest to many people for obvious reasons. One way of predicting future behavior is by studying past behavior. For example, consider the following rise and fall of a stock over a period of 15 days. Positive values indicate the amount by which the stock rose on that day and negative values indicate the amount by which the stock fell.

   -6 -1 +5 + 7 +5 +3 -5 -7 -1 +10 +1 +5 -20 +10 +1

   We want to determine what the best days were to buy and sell this stock. You are allowed to buy only once and sell only once during the 15 day period.

   For example, if you bought on day 3 and sold on day 6, then you would have a net profit of $20. However, if you bought on day 3 and sold on day 12, then you would have a net profit of $23.

You developed a brute-force solution to this problem previously. Now, give pseudocode for a divide-and-conquer algorithm that computes the best day to buy and the best day to sell given the stock's history over the past n days. Assume that the rise and fall values are stored in an array A[1...n]. Give the Θ efficiency class of your algorithm (it should be better than that from brute force!) and briefly explain how you arrived at this bound. (6 points)

5. Baseball teams have reported for spring training, so let's consider a problem involving baseball statistics. Suppose you have a large array of numbers representing the batting averages of all players in a league who played in some minimum number of games, *e.g.*:

$$A = .312, .288, .185, .300, .276, .297, .307, .330.$$

Your task is to find the pair of batting averages that are closest together in value. Develop pseudocode for an algorithm to solve this problem the operates in better than $O(n^2)$ time. Give the Θ efficiency class of your algorithm and briefly explain how you arrived at this bound. (5 points)

Submitting

Before 11:59 PM, Thursday, March 18, 2015, submit your problem set for grading. To complete the submission, upload an archive (a .7z or .zip) file containing all required files using Submission Box at http://sb.teresco.org under assignment "PS5".

Grading

This assignment is worth 55 points, which are distributed as follows:

Feature	Value	Score
Merge sort implementation	5	
Merge sort section of empirical analysis	6	
Quicksort implementations	10	
Quicksort section of empirical analysis	12	
Exercise 5.2.5	3	
Exercise 5.2.6	3	
Quickhull drawings	5	
Divide-and-conquer stock algorithm	6	
Batting averages algorithm	5	
Total	55	