

Homework 3*Instructor: Shi Li***Deadline: 3/25/2020**

Your Name: _____ Your Student ID: _____

Problems	1	2	3	Total
Max. Score	10	20	20	50
Your Score				

Problem 1 (10 points). For each of the following recurrences, use the master theorem to give the tight asymptotic upper bound.

1. $T(n) = T(n/2) + O(n)$.
2. $T(n) = 4T(n/2) + O(n^2\sqrt{n})$.
3. $T(n) = 9T(n/3) + O(n^2)$.
4. $T(n) = 5T(n/2) + O(n^2)$.

Problem 2 (20 points). We consider the following problem of counting stronger inversions. Given an array A of n positive integers, a pair $i, j \in \{1, 2, 3, \dots, n\}$ of indices is called a strong inversion if $i < j$ and $A[i] > 2A[j]$. The goal of the problem is to count the number of strong inversions for a given array A .

Give an $O(n \log n)$ -time divide-and-conquer algorithm to solve the problem.

Problem 3 (20 points). Suppose you know the prices of one stock in sequence of days. You can buy one share of the stock in some day and sell it in another day later. The goal of the problem is to maximize the profit. Notice that you can only buy and sell the stock once. If you can not make a profit, you do not need to buy and sell the stock.

For example, if the prices of the stock in a sequence of 7 days is 15, 30, 18, 45, 9, 40, 35. Then you can buy one share of the stock in day 5 and sell it in day 6 and then your profit is $40 - 9 = 31$.

Design an $O(n \log n)$ -time algorithm to solve the problem.