

Due at the start of class Thursday, September 26, 2002.

Problem 1. Problem A.1-4 (page 1062) NEW EDITION
Problem 3.1-3 (page 45) OLD EDITION

Problem 2. Problem A.1-6 (page 1062) NEW EDITION
Problem 3.1-5 (page 45) OLD EDITION

Problem 3. Recall the algorithm described in class on Sep 12th (Thu) that given n numbers stored in sorted order, checks to see if there exist two numbers in the set that add to x . (The same number cannot be used twice.)

First write out a pseudo-code description of the algorithm.

Now give a proof that the worst case running time is $O(n)$.

Problem 4. Use the iteration method to solve the following recurrence.

$$T(1) = 1$$

$$T(n) = T(n - 1) + n^2 \quad \text{if } n > 1$$

Find a function $g(n)$ such that $T(n) = \Theta(g(n))$. (The function $g(n)$ should be expressed without the use of any \sum symbols.)