

## CS / MCS 401 Week #3 Exercises (Fall 2007)

**Exercise 2.3–1** (*page 36*).

**D.** Prove by induction on  $n$  that the recurrence

$$C(n) = d + C(k) + C(n-k-1), \quad C(0) = 0,$$

where  $d$  is a constant and  $0 \leq k \leq n-1$ , has solution  $C(n) = dn$ .

**E.** Show how to merge three sorted arrays of length  $n/3$  into a single sorted array of length  $n$  using approximately  $5/3n$  comparisons, in the worst case.

**F.** Consider 3-way mergesort, in which an array of size  $n$  is divided into three subarrays of equal size, and the three subarrays are sorted by recursive calls to 3-way mergesort. The recurrence for the number of comparisons becomes  $C(n) = 3C(n/3) + (5/3)n$ ,  $C(1) = 0$ , assuming  $n$  is a power of 3, say  $n = 3^k$ . Find an exact solution to this recurrence, when  $n$  is a power of 3. (Your final solution should involve only  $n$ , not  $k$ .) How does the number of comparisons performed by 3-way mergesort compare to the number performed by ordinary mergesort?