Purdue University
CS 381

S. Hambrusch - C. Guerra
Fall 1999

Assignment 5

Due: Friday, October 15 1999, in class

Remember: *The midterm exam is on Tuesday, October 19, 8:30-9:30pm in SMTH 108! You need to bring your student ID to the exam.*

1) (15 pts.) Assume you are given a sequence of n elements $a_1, a_2, \cdots, a_n$ to be sorted. The sequence has the property that the location of element a_i in the sorted sequence is at most d positions away from position i; i.e., if a_i ends up at position j, then $|i-j| \leq d$. Describe and analyze an algorithm to sort the sequence $a_1, a_2, \cdots, a_n$ in $O(n \log d)$ time when you know the value of d. Hint: Use a heap as a data structure.

2) (15 pts.) Consider the following modification to the base case of *Mergesort* when sorting an array A of size n. Given is a parameter k, $1 \leq k \leq n$; when the array to be sorted has size less than or equal to k, insertion sort is invoked. This corresponds to ending the recursion of Mergesort when the array size is $\leq k$.

(a) What is the worst-case time complexity of this sorting algorithm?

(b) What is the largest value of k as a function of n that gives the same time asymptotic complexity as the standard mergesort?

3) (20 pts.) (i) Consider a heap of size n. Describe an efficient algorithm for finding the 4-th largest element.

(ii) Let S be a set of n elements which contains only 8 distinct elements. You do not know these 8 elements. Describe and analyze an efficient algorithm to sort set S under this assumption.