CSCI 539 Algorithms

Homework 4

Due: November 8, 2001

1. Determine the time complexity of Insertion Sort, Selection Sort, Bubble Sort, Heap Sort, Merge Sort, and Quick sort on sorted input, reversely sorted input, and input of identical elements. Assume that the input list has n elements. Assume that in Quick Sort the pivot is the first element of the list to be sorted. You may provide the answers by filling out the following table.

	Sorted	Reversely sorted	Identical
Insertion Sort			
Selection Sort			
Bubble Sort			
Heap Sort			
Merge Sort			
Quick Sort			

2. Suppose you are given a sorted list of $n-k$ elements followed by k randomly order elements. How would you sort the entire list if

   (a) $k = \Theta(1)$?

   (b) $k = \Theta(\log n)$?

   (c) $k = \Theta(\sqrt{n})$?

3. Professors Howard, Fine, and Howard have proposed the following "elegant" sorting algorithm:

```
Stooge-sort(A,i,j);
  if A[i] > A[j] then exchange $A[i]$ and $A[j]$;
  if i+1 >= j then return;
  k = (j-i+1) div 3;  //integer division
  Stooge-sort(A,i,j-k);
  Stooge-sort(A,i+k,j);
  Stooge-sort(A,i,j-k);
```

   (a) Argue that `Stooge-sort(A,1,n)` correctly sorts `A[1...n]` by using induction.

(b) Give a recurrence for the worst-case time of `Stooge-sort(A,1,n)` and solve it in Θ.

(c) Is the algorithm any better than the existing $\Theta(n \log n)$ sorting algorithms such as Heap Sort and Merge Sort? Do the professors deserve tenure because of this algorithm?

4. Let $S_1, S_2, \ldots, S_k$ be sets of integers all in the range of 1 to n. Assume the sum of the cardinalities of the S_i's is n. Describe an $O(n)$ algorithm to sort all of the S_i's separately. (Hint: You may want to use Radix Sort, a linear-time algorithm that sorts integers.)