Due: 6 pm on Wednesday, May 7

Assignment 10

1 Matching Nuts and Bolts

We are given n bolts and n nuts of different sizes, where each bolt exactly matches one nut. Our goal is to find the matching nut for each bolt. The nuts and bolts are so similar that you cannot compare two nuts, or two bolts directly; however, we can test whether any nut is bigger than, smaller than, or the same size as any bolt.

(a) Prove that in the worst case, $\Omega(n \log n)$ nut-bolt tests are required to match up all the nuts and bolts.

(b) Prove that in the worst case, $\Omega(n + k \log n)$ nut-bolt tests are required to find k arbitrary matching pairs. (*Hint: Use an adversary argument for the* $\Omega(n)$ *term.*)

2 Almost Sorted!

We say that an array $A[1 \ldots n]$ is k-sorted if it can be divided into k blocks, each of size n/k (you can always assume that n/k is an integer), such that the elements in each block are larger than the elements in earlier blocks, and smaller than elements in later blocks. The elements within each block need not be sorted.

For example, the following array is 4-sorted:

1	2	4	3	7	6	8	5	10	11	9	12	15	13	16	14

(a) Describe an algorithm that k-sorts an arbitrary array in $O(n \log k)$ time.

(b) Prove that any comparison-based k-sorting algorithm requires $\Omega(n \log k)$ comparisons in the worst case.

(c) Describe an algorithm that completely sorts an already k-sorted array in $O(n \log(n/k))$ time.

(d) Prove that any comparison-based algorithm to completely sort a k-sorted array requires $\Omega(n \log(n/k))$ comparisons in the worst case.