CS2223
HW#2

DUE: Monday, November 6

1. (6 points) For each of the following, tell whether or not it is true and justify your response.

a $2^{1.5n} \in O\left(3^n\right)$.

b $3^n \in O\left(2^{1.5n}\right)$.

2. (5 points) Describe an algorithm to find the $20^{\underline{th}}$ smallest element of an array $A[1..n]$ such that in the best case the algorithm uses n-1 pairwise comparisons.

3. (10 points) Assume that $A[1..n]$ is an array of 0s and 1s. The problem is to sort the array, but only *local transpositions* may be used in sorting. That is, the only benchmark operation that may be performed is to "$\text{exchange } A[i] \text{ and } A[i+1]$" for some $1 \le i < n$. Establish a $\Omega\left(n^2\right)$ lower bound on the worst case complexity of this problem. Note that a $\Omega\left(n^2\right)$ lower bound on the worst case complexity of local transposition sorting does not necessarily apply here since we have the special case that there may only be two distinct values in each $A[i]$.

CS2223
HW#2 SOLUTIONS

1. We note that $2^{1.5n} = \left(2^{1.5}\right)^n \approx 2.828427^n < 3^n$ for all $n \geq 1$.

a $2^{1.5n} \in O\left(3^n\right)$. Choose $c = n_0 = 1$ and we note that for all $n \geq n_0$, $2^{1.5n} < 3^n$.

b $3^n \notin O\left(2^{1.5n}\right)$ Assume that $3^n \in O\left(2^{1.5n}\right)$**.** Then there must exist c, n_0 such that for all $n \geq n_0$, $3^n < c2^{1.5n} \approx c * 2.828427^n$. Dividing both sides by 2.828427^n yields $1.06066^n < c$ for all $n \geq n_0$. This is impossible, so, by contradiction, $3^n \notin O\left(2^{1.5n}\right)$.

2. *Ordered?* ← true

   for $i \leftarrow 1$ **to** *n*-1 **do if** $A[i] > A[i+1]$ **then** *Ordered?* ← false

   if *Ordered?* **then return** *A*[20] ▹ *n*-1 comparisons have been made

   INSERTIONSORT(*A*)

   return *A*[20]

3. Consider the array *A* with *n*/2 1's followed by *n*/2 0s. Any local transposition fixes an *inversion* (a 0 and a 1 which are out of order). But *A* has $\left(\frac{n}{2}\right)^2 = \frac{n^2}{4}$ inversions. Hence, any algorithm to sort *A* must use at least $\Omega\left(n^2\right)$ local transpositions.