CS2223
MIDTERM EXAM

Name____________________

Date: November 20, 2003
All documentation permitted

1  ________

2  ________

3  ________

4  ________

TOTAL  ________
1. (25 points) Suppose that you are given an array $A[1..n]$ of numbers, and you seek a maximally distant pair of numbers in $A$. That is, you seek $i$ and $j$, $1 \leq i, j \leq n$ such that for all $k$ and $l$, $1 \leq k, l \leq n$,


Show how to solve this problem in worst case time in $O(n)$. 

2. (25 points) Suppose that you are given an array $A[1..n]$ of integers, which is partially sorted in the sense that there exists an integer $m$, $1 \leq m \leq n$, such that $A[1] \leq A[2] \leq \ldots \leq A[m]$ and $A[m+1] \leq A[m+2] \leq \ldots \leq A[n]$. However, you do not know the value of $m$. Show that $A$ can be sorted in worst-case time in $O(n)$. 
3. (25 points) Assume we are given as input a binary array $A[1..n]$, that is, $A[i] \in \{0,1\}$ for all $1 \leq i \leq n$.

a. Prove an upper bound on the complexity of sorting $A$ by showing that it is in $O(n)$.

b. Prove a lower bound on the complexity of sorting $A$ by showing that it is in $\Omega(n)$. 
4. (25 points) Prove or give a counterexample to the following.

**CONJECTURE:** For any connected graph \( G = (V, E) \) with \( |V| \geq 3 \) and one-to-one function \( w: E \to \mathbb{R}^+ \) which associates a unique weight with every edge \((e \neq e' \to w(e) \neq w(e'))\), the two edges of minimum weights belong to a minimum spanning tree of \( G \).
1. This problem is solved when \( A[i] \) is the minimum element of \( A \) and \( A[j] \) is the maximum element. Both of these values can be computed in linear time.
   - Set \( i \) to index of minimum element
     \( i ← 1 \)
     \( \text{for } k ← 2 \text{ to } n \text{ do} \)
     \( \quad \text{if } A[k] < A[i] \text{ then } i ← k \)
   - Set \( j \) to index of maximum element
     \( j ← 1 \)
     \( \text{for } k ← 2 \text{ to } n \text{ do} \)
     \( \quad \text{if } A[k] > A[j] \text{ then } j ← k \)

2. \( m ← 1 \)
   - Find \( m \)
   \( \text{repeat } m ← m + 1 \text{ until } (A[m] > A[m + 1] \vee m = n - 1) \)
   \( \text{if } m < n \text{ then MERGE}(A[1..m], A[m + 1..n]) \)

3. \( a \) We establish the upper bound by providing a linear time algorithm.
   \( \text{num0s ← 0} \)
   - count the 0s in \( A \)
   \( \text{for } i ← 1 \text{ to } n \text{ do} \)
     \( \quad \text{if } A[i] = 0 \text{ then } \text{num0s ← num0s + 1} \)
   - fill in the 0s, then the 1s in \( A \)
   \( \text{for } i ← 1 \text{ to num0s} \text{ do} \)
     \( \quad A[i] ← 0 \)
   \( \text{for } i ← \text{num0s} + 1 \text{ to } n \text{ do} \)
     \( \quad A[i] ← 1 \)

\( b \) To establish a linear time lower bound, we note that if an algorithm existed to sort \( A \) without examining some element, say \( A[i] \), then we could change \( A[i] \), leaving the other \( n-1 \) elements of \( A \) unchanged, and the algorithm would have to yield the same result, which is now incorrect.

4. The CONJECTURE is true. Neither edge can introduce a cycle (two edges can’t form a cycle) when considered by Kruskal’s Algorithm, so each would be added to the minimum spanning tree.