CSCC73
Algorithm design & analysis

Week 6 Tutorial
Question 1

Recall that $z$ is a **primitive** $n$-th root of 1 if
(a) $z^n = 1$; and
(b) $z^k \neq 1$ for all $k$, $0 < k < n$.
Thus, if $z$ is a primitive $n$-th root of 1, then $z^0, z^1, z^2, \ldots, z^{n-1}$ is the list of **all** the $n$-th roots of 1.

Prove the following facts (used in our discussion of the FFT):
• If $z$ is a primitive $n$-th root of 1, then $z^{-1}$ is also a primitive $n$-th root of 1.
• If $z$ is a primitive $n$-th root of 1, then $z^2$ is a primitive $n/2$-th root of 1.
Question 2

Demonstrate the polynomial multiplication algorithm based on the FFT by using it to multiply the polynomials $x^2+1$ and $x+1$.

- Use the FFT to evaluate each polynomial at an appropriate set of points, and show the resulting vectors;
- multiply the vectors component-wise; and
- use the inverse FFT to obtain the coefficients of the final result.
Question 3

Minimum spanning tree problem: Given an undirected, connected graph with edge weights, find a subset of the edges that form a tree with minimum weight.

Proposed D&C MST algorithm:

• If the graph has one node, return the empty tree
• Otherwise,
  – partition the set of nodes of the graph into any two sets $V_1$ and $V_2$ of about the same size
  – recursively find MSTs of the graphs induced by $V_1$ and $V_2$, and join them by the minimum weight edge connecting a node in $V_1$ to a node in $V_2$

Does this algorithm work?
Question 4

Suppose we are given an array $A[1..n]$ of numbers in arbitrary order, and an integer $k$, $1 \leq k \leq n$.

Give an algorithm that returns the $k$ largest elements of $A$ (in any order).

Your algorithm must run in $O(n)$ time.
Question 5

Suppose we are given two sorted arrays of distinct numbers $A[1..n]$ and $B[1..n]$.

Give an algorithm that finds the median of the union of the two arrays.  
Your algorithm must run in $O(\log n)$ time.