Problems:

1. QuickSelect is the following simple algorithm for finding the $k$-th smallest element in an unsorted set $S$.

   \[
   \text{QuickSelect}(S, k):
   \]
   
   (a) Pick a pivot element $p$ uniformly at random from $S$.
   
   (b) By comparing $p$ to each element of $S$, split $S$ into two pieces: $S_1 = \{x \in S | x < p\}$ and $S_2 = \{x \in S | x > p\}$
   
   (c) If $|S_1| = k - 1$ then output $p$
   
   If $|S_1| > k - 1$, then output QuickSelect($S_1$, $k$)
   
   If $|S_1| < k - 1$, then output QuickSelect($S_2$, $k - |S_1| - 1$)

   Prove the best bound you can on the expected number of comparisons made by QuickSelect on a set $S$ of size $n$. You may assume that initially $k = n/2$ (i.e., we are trying to find the median of $S$) which is the worst case.

2. Generalizing the notion of a cut-set, we define an $r$-way cut-set in a graph as a set of edges whose removal breaks the graph into $r$ or more connected components. Explain how the basic randomized min-cut algorithm (not the recursive version) can be used to find minimum $r$-way cut sets, and bound the probability it succeeds in one iteration. How many repetitions of a complete iteration would be needed to reduce the probability of error to $1/n$.

3. Think about and explain as best you can the following design decisions from the linear time randomized minimum spanning tree algorithm:

   - the decision to do two Boruvka steps at the beginning (as opposed to say 0, 1 or more than 2 Boruvka steps).
   
   - the decision to sample half the edges as opposed to a fraction $p$ of the edges for some choice of $p \neq 1/2$. 
