CSE 521: Design and Analysis of Algorithms
Assignment #7
Due: Wednesday, Feb 25

Problems: do any 3 of the following 4 problems. You can choose to do a 4th problem for extra credit – if you do, please label which one you want to count as extra credit.

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

   `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. You are watching a stream of packets go by one at a time, and want to take a random sample of k distinct packets from the stream. You face several problems:

   - You only have room to save k packets at any one time.
   - You do not know the total number of packets in the stream.
   - If you choose not to save a packet as it goes by, it is gone forever.

   Devise a scheme so that, whenever the packet stream terminates, you are left holding a subset of k packets chosen uniformly at random from the entire packet stream. If the total number of packets in the stream is less than k, you should hold all of these packets.

3. Give a Monte Carlo algorithm that finds the *second smallest cut* in an undirected graph on n vertices in $O(n^2 \log^3 n)$ time with high probability (where high probability means probability at least $1 - n^{-c}$ for some constant $c > 0$). Note that if a graph has two distinct minimum cuts, the second smallest cut is a minimum cut. (A randomized algorithm is *Monte Carlo* if there is a low probability, in this case $O(n^{-c})$, of producing an incorrect answer.)

4. 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 $p \neq 1/2$.