Homework 6
CS 3385

1. What is the running time (using asymptotic notation) of the `insert()` operation for a heap-based priority queue? Briefly justify your answer.

2. Using figure 7.1 as a model, illustrate the operation of `partition()` on the array $A = \langle 13, 19, 9, 5, 12, 8, 7, 4, 21, 2, 6, 11 \rangle$.

3. In the video we claimed that $T(n) = T(n - 100) + \Theta(n) = \Theta(n^2)$. Show that this is true using the substitution method.

4. Consider an array in which every element is within 3 positions of its sorted position. Thus, the array is *almost* sorted.

   (a) Insertion sort is $O(n^2)$. Using the algorithm at the top of page 26 of the textbook, show the running time of the almost-sorted array.

   (b) Give the recurrence for quicksort on the almost-sorted array. Don't worry about rounding. You can assume that each element is *exactly* 3 spots away from sorted order at each level of the recursion tree. Assume that `partition()` is $\Theta(n)$.

   (c) What is the solution to the recurrence? You do not need to justify your answer.

5. Suppose that the splits at every level of quicksort are in the proportion $1 - \alpha$ to α where $0 < \alpha \leq 1/2$ is a constant. Show that the minimum depth of a leaf in the recursion tree is approximately $-\lg n / \lg \alpha$ and the maximum depth is approximately $-\lg n / \lg(1 - \alpha)$. Don't worry about integer round-off.

6. A random number generator can be expensive. Assuming the randomized version of quicksort splits every level perfectly in half, how many times is `random()` called for input size n? Use asymptotic notation for your final answer.