Algorithm 2016 Fall Homework 1

1. Illustrate the operation of merge sort on the array A = 〈 5, 22, 76, 92, 32, 1, 63, 21 〉.
2. Give asymptotic upper and lower bounds for T(n) in each of the following recurrences. Assume that T(n) is constant for sufficiently small n. Make your bounds as tight as possible, and justify your answers.
   i. $T(n) = 2T\left(\frac{n}{2}\right) + n^3$
   ii. $T(n) = T\left(\frac{9n}{10}\right) + n$
   iii. $T(n) = T(n-1) + n$
3. Argue that the solution to the recurrence T(n) = $T(n/3) + T(2n/3) + cn$, where c is a constant, is $\Omega(n \lg n)$ by appealing to a recursion tree.
4. Can the master method be applied to the recurrence $\mathrm{T}(n) = 4\mathrm{T}\left(\frac{n}{2}\right) + n^2 \lg n$? Why or why not? Give an asymptotic upper bound for this recurrence.
5. Prove $a^{\log_b c} = c^{\log_b a}$.
6. Prove log (n!) = $\Theta(n \log n)$.
7. Show that quicksort's best-case running time is $\Omega(n \log n)$.
8. Show that when all elements are distinct, the best-case running time of HEAPSORT is $\Omega(n \log n)$.
9. Is the sequence (23, 17, 14, 6, 13, 10, 1, 5, 7, 12) a max-heap?
10. Show that there are at most $\left\lceil \frac{n}{2^{h+1}} \right\rceil$ nodes of height h in any n-element heap.