ALGORITHMS, FALL 2019, HOMEWORK 3

– This assignment is worth 2 units.
– Due on Thursday, September 26, at noon.

1. Use the master method for the following (state case or explain what dominates, and state the answer), or explain why it's not possible.

   (a) $T(n) = 10 \cdot T(\frac{n}{3}) + \Theta(n^2 \log^5 n)$.

   (b) $T(n) = 256 \cdot T(\frac{n}{4}) + \Theta(n^4 \log^4 n)$.

   (c) $T(n) = T(\frac{19n}{72}) + \Theta(n^2)$.

   (d) $T(n) = n \cdot T(\frac{n}{2}) + n^{\log_2 n}$.

   (e) $T(n) = 16 \cdot T(\frac{n}{4}) + n^2$.

   (f) $T(n) = 3 \cdot T(\frac{n}{2}) + n^2$.

   (g) $T(n) = T(\frac{n}{n-1}) + 1$.

   (h) $T(n) = 4 \cdot T(\frac{n}{16}) + \sqrt{n}$.

2. In class we learn how to find the number that has rank r among n elements, in $\Theta(n)$ time, using the classic algorithm with groups of size 5.

   (a) Show what happens if we form groups of size 3 instead.

   (b) Show what happens if we use groups of size k, where k represents an odd integer greater than 5.

   (c) Show what happens if we use $\sqrt{n}$ groups of size $\sqrt{n}$.

   (d) Show what happens if we use 5 groups of size $\frac{n}{5}$.

   In all cases, either prove that we still get $O(n)$ time or show that we cannot get $O(n)$ time. You may assume that division of produces an integer value. In other words you may assume that n is a power of whatever group size is used.