CSE 2331 - Problem Set 3
Due beginning of lecture on September 30th

Problem numbers are from the third edition of "Introduction to algorithms". If unsure about which problem to solve, ask. Collaboration is permitted; looking for solutions from external sources (books, the web, etc.) is prohibited.

1. Write a recurrence relation describing the worst case running time of each of the following algorithms and determine the asymptotic complexity of the function defined by the recurrence relation. Justify your solution. Assume that all arithmetic operations take constant time.

   (a)
   ```
   function func3(n)
   if n <= 10 then return(n)
   x = floor(n/7)
   x = x + func3(floor(3n/4))
   return(x)
   ```

   (b)
   ```
   function func4(A,n)
   /* A is an array of n integers */
   if n <= 2 then return (A[1])
   for i = 1 to floor(n/2) do
       A[i] = A[i] + A[i+1]
       B[i] = A[i] - A[i-1]
       C[i] = A[i] + A[2*i]
   endfor
   x = 0
   x = x + func4(A, floor(n/3))
   x = x + func4(B, floor(n/3))
   x = x + func4(C, floor(n/3))
   return(x)
   ```

2. 5.4-2

3. 7.3-2

4. 7.4-4

5. The kth quantiles of an n-element set are the $k-1$ order statistics that divide the sorted set into k equal-sized sets (to within 1). For example, the 4th quantiles of an array of size 64 are the 16th, 32nd and 48th order statistics. Give an $O(n \log k)$-time algorithm to list the kth quantiles of a set.