CSE 331 Algorithms and Data Structures (Section 1) Homework 1

Out on Jan. 20 2015. Due by Tuesday's class on Jan. 27. If you don't want to type, please make an effort to use neat, clean handwriting.

1. Rewrite the INSERTION-SORT procedure to sort into descending instead of ascending order. Use the same pseudocode conventions as presented in lecture. (6 points)

2. Consider the following functions in terms of order of magnitude. Order (list) them in terms of the growth, such that slow growth functions are listed before fast growth functions. That is, if f is listed before g, $f = O(g)$. Also, partition the functions so that if $f = \Theta(g)$, then they are in the same class. (14 pts)

   N
   2/N
   37
   $N!$
   $N \log N$
   $\log(N)$
   2^N
   $N \log^2(N)$
   $N^{1.5}$
   $N^2 \log(N)$
   N^3
   $N \log(N^2)$
   N^2
   $\sqrt{N}$

3. Prove that $3n^2 + 2n \log n + 10 = O(n^2)$. (6 pts)

4. Suppose $T_1(N) = O(f(N))$ and $T_2(N) = O(f(N))$. Evaluate each statement below. If it is true, prove it. If it is not true, provide a counter-example. (12 pts)

   - a. $T_1(N) + T_2(N) = O(f(N))$
   - b. $T_1(N)/T_2(N) = O(1)$
   - c. $T_1(N) = O(T_2(N))$

5. Let $f(n)$ and $g(n)$ be asymptotically nonnegative functions. Using the basic definition of Θ-notation, prove that $max(f(n), g(n)) = \Theta(f(n) + g(n))$.(12 pts)

6. For each of the following four program fragments, give an analysis of the running time. Give this time in Big-O notation using function with the least order of magnitude. Basically don't say the complexity is $O(N^9)$ if it is also $O(N^2)$. (10 pts)

   (a)
   ```
   Sum = 0;
   for i = 1 to N do
       Sum = Sum +i;
   ```

   (b)
   ```
   Sum = 0;
   for i = 1 to N do
       for j = 1 to N do
           for k = 1 to N do
               Sum = Sum + 1;
   ```

   (c)
   ```
   Sum = 0;
   for i = 1 to N^2 do
       for j = 1 to N do
           Sum = Sum + 1;
   ```