CSCI 533 Analysis of Algorithms
Homework #1
Due Tuesday, August 29th at Midnight

Most of this material should be a review of your undergraduate algorithms course. Please follow the posted guidelines for completing homework assignments. If you make any assumptions, please state them. 10 points per question.

1. List the following functions in order by their Big-Oh notation. If two or more are of the same complexity, clearly indicate it by putting them on the same line. Note: You do NOT have to show proofs for these.

- $n \log n$
- $3n - 12$
- $n!$
- $5n^2 + 2n$
- $4n - 6$
- $\sqrt{n}$
- $n^n - 4 \log n$
- $n \ln n$
- $\log \log n$
- $5n^3 + 3n^2 + 16n - 7$
- $(\log n)^2$ (aka $\log^2 n$)
- $2 \log n + 3 \log \log n$

2. For each of the following pairs of functions mark the best answer of:

(a) $f(n) \in O(g(n))$
(b) $f(n) \in \Omega(g(n))$
(c) $f(n) \in \Theta(g(n))$

- $f(n) = \log n^2$  $g(n) = \log n + 5$
- $f(n) = \sqrt{n}$  $g(n) = \log n$
- $f(n) = \log \log n$  $g(n) = \log n$
- $f(n) = n + n \log n$  $g(n) = \log n$
- $f(n) = 2^n$  $g(n) = 3^n$

3. What is the time complexity of each of the following code segments?

(a) 
```plaintext
i = n
while i > 0
    for j = i to n
        // constant work
    endfor
i = i - 1
endwhile
```
\begin{verbatim}
(b)  i = 1
    while i <= n
        j = n
        while j >= 1
            // constant work
            j = floor (j/2)
        endwhile
        i = i + 1
    endwhile
\end{verbatim}

4. Graphs

(a) How many edges does an undirected graph have if it has vertices of degree 4, 3, 3, 2 and 2? Draw a graph that meets the criteria. Label the vertices A, B, C, D and E.

(b) Construct an adjacency matrix representation for your graph.

(c) Construct an adjacency list representation for your graph.

(d) In general, how much storage is needed to represent a directed graph of \( n \) vertices and \( m \) edges for each of the following cases? Do NOT use Big-Oh notation. Assume pointers, values, weights, and vertex names each require 1 word of storage. Explain your answer (a drawing may help).
   
i. adjacency lists for unweighted graph?
   ii. adjacency lists for weighted graph?
   iii. adjacency matrix for unweighted graph?
   iv. adjacency matrix for weighted graph

5. Suppose you are given a sorted list of \( n \) elements and you want to determine whether there is a pair of elements \( a \) and \( b \) in the list such that \( a + b = X \) for some given value of \( X \). Design an \( O(n \log n) \) algorithm to determine whether such a pair exists. Be sure to make the algorithm clear, argue its correctness and analyze its complexity.

6. Graduate students only. Repeat the previous problem, but this time design an \( O(n) \) algorithm.