COP4531: Midterm Practice Questions

**Question 1** State True or False and prove your answer using only the definitions for $o, O, \omega, \Omega, \Theta$:

A. if $f(n) \in \omega(g(n)) \rightarrow f(n) \in \Omega(g(n))$

B. if $f(n) \notin \omega(g(n))$ and $f(n) \notin o(g(n)) \rightarrow f(n) \in \Theta(g(n))$

C. if $f(n) \in \omega(g(n)) \rightarrow f(n) \notin O(g(n))$

D. if $f(n) \in O(g)$ and $f(n) \in \Omega(g(n))$ then $f(n) \in \Theta(g(n))$.

**Question 2** Prove that the following statements are either True or False:

A. Given any polynomials $f_1(x), f_2(x), f_3(x)$, $\Theta(f_1 + f_2 + f_3) = \Theta(MAX(f_1, f_2, f_3))$.

B. Given any polynomials $f_1(x), f_2(x), f_3(x)$, $\Theta(f_1 \cdot f_2 \cdot f_3) = \Theta(MAX(f_1, f_2, f_3))$.

**Question 3** Formally prove the following statements.

A. $0.2n^4 \in \Theta(4n^4)$.

B. $0.5 \cdot 2^n \log n \in \omega(n^5)$.

C. If $f \in \Omega(g)$ and $f' \in \Omega(g')$, then $ff' \in \Omega(gg')$.

**Question 4** Given a large data set, what questions would you ask and how would they help you determine which of the following sorting algorithms you would use.

A. Quick Sort

B. Merge Sort

C. Insertion Sort

**Question 5** Give the running time of the following code segments. For all the questions below, you may assume that $n$ is the length of an input array.

A. i = 0
   while(True)
   i = i + 2
   if(i>(10^10)^{10})
   return n
B. $f(A)$:
   if $A$.length $< 2$:
       return $A[1]$
   else:
       return $\text{MIN}(f(A[1..n/2-1]), f(A[n/2..n]))$

C.
   $i = n^3$
   while $i > 1000$:
       $i = i/2$

Question 6
   for $i = 2$ to $n$
       $j = i$
       while $j > 1$ and $A[j-1] > A[j]$
           swap $A[j]$ and $A[j-1]$
           $j = j - 1$
       end while
   end for

A. What does the above code do?
B. Give the best case running time for the above code segment.
C. Give the worst case running time for the above code segment.

Question 7 Leonardo numbers are defined as follows:
   - $L(0) = 1$
   - $L(1) = 1$
   - $L(n > 1) = L(n-1) + L(n-2) + 1$

A. Give the recurrence relation, $T(n)$ for Leonardo numbers.
B. Give pseudo code for the Dynamic Programming solution to finding Leonardo numbers and justify your code’s run time.

Question 8 Prove or disprove the following statement: For the rod-cutting problem, the greedy solution is optimal if only prime cut-lengths are allowed.

F: $S = \{s_1, s_2, s_3, \ldots, s_k\}$, M:
   Sort $S$
   for $i = k$ to 1
       print $S[i]$, M/$S[i]$
       $M = M\times S[i]$

Question 9 Justify why Lloyds’ method must be run multiple times.
**Question 10** Draw a simple data set and a set of initial centers on which Lloyd’s method can fail, give the clustering given by Lloyds method and the optimal k-means clustering.