Fall 2013 CSE310 Midterm 1A (in class)

Instructions:

- There are five problems in this paper. Please use the space provided (below the questions) to write the answers.
- Budget your time to solve various problems (roughly 15 minutes for each problem) and avoid spending too much time on a particular question.
- This is a **closed book** examination. You may not consult your books/notes. Cell phones and computers are not allowed. However, you can use a basic calculator.
- You are **NOT** supposed to use a pencil. If you use a pencil, you cannot challenge your grade after the midterm is graded.

NAME	
ASUID	

Question	Score
P1	
P2	
P3	
P4	
P5	
Total	

Problem 1A. (20 points: 4 + 4 + 4 + 4 + 4)

Use O, Ω, or Θ to relate each of the following pairs of functions. In particular, you need to write your answer as $f(n) \in O(g(n))$, $f(n) \in \Omega(g(n))$, or $f(n) \in \Theta(g(n))$. In case $f(n) \in \Theta(g(n))$, you will not receive credit if you write $f(n) \in O(g(n))$ or $f(n) \in \Omega(g(n))$, although both are implied by $f(n) \in \Theta(g(n))$.

1. $f(n) = 100 \times \log n$, $g(n) = 0.001 \times \sum_{i=1}^{n} i$.
   Answer: $f(n) \in O(g(n))$.

2. $f(n) = 1.0001^n$, $g(n) = n^{300}$.
   Answer: $f(n) \in \Omega(g(n))$.

3. $f(n) = \sum_{i=1}^{n} i^2, g(n) = n^3 + \sum_{i=1}^{n} i$.
   Answer: $f(n) \in \Theta(g(n))$.

4. $f(n) = 2 \times n!, g(n) = (2 \times n)!$.
   Answer: $f(n) \in O(g(n))$.

5. $f(n) = n^{0.1}, g(n) = \sum_{i=1}^{n} \frac{100}{i}$.
   Answer: $f(n) \in \Omega(g(n))$.

Problem 2A. (20 points: 5 + 5 + 5 + 5)

There are two algorithms A_1 and A_2, with time complexities $T_1(n)$ and $T_2(n)$, respectively. We know that $T_1(n) = 7T_1(n/2) + 1000n^2$; $T_2(n) = 8T_2(n/2) + n^2$;

1. Use the master method to decide the asymptotic notation of $T_1(n)$. You may use $\log_2 7 = 2.81$ in your calculations.

   Answer: $a = 7, b = 2, f(n) = 1000n^2$. $\log_b a = 2.81$. Take $\epsilon = 0.1$, then $f(n) \in O(n^{(\log_b a)-\epsilon})$. This is case 1 of the master method. So $T_1(n) \in \Theta(n^{2.81})$.

   Grading: 5 pts for the correct answer. If the answer is wrong, earn 2 pts for knowing the correct case.

2. Use the master method to decide the asymptotic notation of $T_2(n)$. You may use $\log_2 8 = 3$ in your calculations.

   Answer: $a = 8, b = 2, f(n) = n^2$. $\log_b a = 3$. Take $\epsilon = 0.1$, then $f(n) \in O(n^{(\log_b a)-\epsilon})$. This is case 1 of the master method. So $T_1(n) \in \Theta(n^3)$.

   Grading: 5 pts for the correct answer. If the answer is wrong, earn 2 pts for knowing the correct case.

3. Which algorithm is asymptotically faster?

   Answer: A_1 is faster.

   Grading: 5 pts for the correct answer. No credit for saying T_1 is faster or T_2 is faster.

4. The recurrence $T(n) = 7T(n/2) + n^2$ describes the running time of an algorithm A. A competing algorithm A' has a running time of $T'(n) = aT'(n/4) + 10n^2$. What is the largest integer value for a such that A' is asymptotically faster than A? You have to write down the value of a, and a brief explanation.

   Answer: Using the master method, we know that $T(n) \in \Theta(n^{\log_2 7})$. If $a = 49$, then $T'(n) \in \Theta(n^{\log_2 7})$.

   For any $a > 49$, we have case 1 and $T'(n) \in \Theta(n^{\log_4 a})$, and hence $T'(n) \in \Omega(n^{\log_2 7})$. For $a = 48$, we have case 1 and $T'(n) \in \Theta(n^{\log_4 48})$. Since $\log_4 48 < \log_4 49 = \log_2 7$, $a = 48$ is the largest integer value for a such that algorithm A' is asymptotically faster than A.

   Grading: 5 pts for the correct answer. If the answer is wrong, earn 3 pts for writing $a = 49$.