# CS 5200 Midterm Practice

100 pts. Duration: 75 mins

Submit your answers in a single PDF file.

**Problem 1 ($5 \times 5 = 25$ pts):** Determine whether the following statements are **true** or **false**. (In the following assume $n$ is a non-negative integer and all functions are from the non-negative integers to the positive reals.) (no need to justify your answer)

1. $n^3 = O((\log n)^{\sqrt{\log n}})$

2. If $f(n) = \Omega(g(n))$ and $g(n) = O(h(n))$, then $f(n) = \Omega(\min(g(n), h(n)))$

3. If $S(n) = 4S(\sqrt{n}) + \log n$ for all $n \geq 2$ and $S(1) = 1$, then $S(n) = O(\log^2 n)$.

4. It is feasible for a function $f(n)$ to be $O(n^2)$ when $g(n) = n^3$ and $h(n) = \sqrt{n}$, yet not be $O(g(n) + h(n))$.

5. If $f(n) = 2^n + o(2^n)$, then $f(n) \neq \Theta(2^n)$.

**Problem 2 (25 pts)** Using the substitution method (do not use Master theorem) solve the recurrence relation $T(n) = 2T(n/3 + 2) + O(n)$, assume $T(1) = c$ is some constant.

**Problem 3 (25 pts)** Give a polynomial-time algorithm for computing $a^{b^c} \mod p$, given $a, b, c$ and $p$. You may assume all integers are $\Theta(\log n)$ bits long.

**Problem 4 (25 pts)** Consider the problem of computing $N! = 1 \cdot 2 \cdot 3 \cdots N$.

(a) If $N$ is an $n$-bit number, how many bits long is $N!$, approximately (in $\Theta(\cdot)$ form)?

(b) Give an algorithm to compute $N!$ and analyze its running time. You may assume operations involving constant number of bits takes constant time.