CSCI2100/ESTR2102: Quiz 1

Hand-write all your solutions on paper. Take a picture of the paper **together with** your CUHK student ID. Upload the picture to Blackboard or email it to the instructor at taoyf@cse.cuhk.edu.hk. Your must do so within 15 minutes after the quiz has started.

Problem 1 (20%). Prove $100n + 10000 = O(n^{1+c})$ where c is the last digit of your student ID.

Solution. $100n + 10000 \leq 10100n^{1+c}$ for all $n \geq 1$.

Problem 2 (50%). Prove that n^{2+c} is not $O(n)$ where c is the last digit of your student ID.

Solution. Our proof here assume $c = 0$. Assume, for contradiction purposes, that $n^2 = O(n)$, namely, there exist constants c_1, c_2 such that $n^2 \leq c_1 n$ for all $n \geq c_2$. This means $n \leq c_1$ for all $n \geq c_2$, which is impossible and, hence, gives a contradiction.

Problem 3 (30%). Suppose that A is a sorting algorithm with worst case running time $\Theta(n^{2+c})$, where c is the last digit of your student ID. Someone claims that merge sort has smaller running time than A on every possible input. Is the claim correct? Why?

Solution. No. Let S be the set of all possible inputs of size n. Given an input $I \in S$, define $cost_I(A)$ as the running time of A on I. From the fact that A has worst case running time $\Theta(n^{2+c})$, we know:

$$\max_{I \in S} cost_I(\text{merge sort}) \quad \leq \quad \max_{I \in S} cost_I(A).$$

But this does not mean $cost_I(\text{merge sort}) \leq cost_I(A)$ for every I.