Written Assignment #1 Due: Oct 5 by 15:40pm

1. (20%) Let $f(n) = O(n)$ and $g(n) = O(n^2)$. Prove that:
   (a) $f(n) + g(n) = O(n^2)$.
   (b) $f(n) \cdot g(n) = O(n^3)$.

2. (30%) Give asymptotic upper bounds for the following $T(n)$, and justify your answers. You may assume that $T(n)$ is constant for $n \leq 20$.
   (a) $T(n) = 4T(\lfloor n/2 \rfloor) + n^2\sqrt{n}$.
   (b) $T(n) = 27T(\lceil n/3 \rceil) + n^3$.
   (c) $T(n) = T(n - 13) + n$.

3. (20%) Let A be an array of n real numbers. Devise an algorithm to determine whether there exists a number in A that appears more than $n/2$ times.
   (a) Give the pseudocode of your algorithm for decide(A, n) so that decide(A, n) outputs "Yes" if there exists such a number in A, or otherwise outputs "No."
   (b) Analyze the running time of your algorithm.
   (c) Prove or disprove that this problem can be solved in $o(n \log n)$ time in the comparison-based model.

4. (15%) Goldbach's conjecture states that every even integer $k > 2$ can be expressed as the sum of two primes. Devise an $O(n)$-time algorithm to determine whether the conjecture holds for a given even integer $k \leq n$. You may assume that you have an array S of primes $\leq n$ where $S[i]$ is the i-th prime and S has length $O(n/\log n)$.
   (a) Give the pseudocode of your algorithm for test_GC(k) so that test_GC(k) outputs "Yes" if Coldbach's conjecture holds for k, or otherwise outputs "No."
   (b) Analyze the running time of your algorithm.
   (c) Does your algorithm run in $o(n)$ time? Justify your answer.

5. (15%) Let S be an array of n distinct integers where $S[1] < S[2] < \cdots < S[n]$. Devise an $O(\log n)$-time algorithm to determine whether there exists $S[i] = i$ for some index i.
   (a) Give the pseudocode of your algorithm.
   (b) Prove the correctness of your algorithm.
   (c) Show that your algorithm runs in $O(\log n)$ time.

if, S[i] > i, 那 S[i] 之後的 element 都不可能符合惹，因為 S 內每個數字都不一樣。