Chapter 2

Asymptotic notations

1. $f(n) = \Theta(g(n))$ and $g(n) = \Theta(h(n))$ imply $f(n) = \Theta(h(n))$
2. $f(n) = o(g(n))$ and $g(n) = o(h(n))$ imply $f(n) = o(h(n))$
3. **Chip testing**
   Professor Diogenes has n supposedly identical integrated-circuit chips that in principle are capable of testing each other. The professor's test jig accommodates two chips at a time. When the jig is loaded, each chip tests the other and reports whether it is good or bad. A good chip always reports accurately whether the other chip is good or bad, but the professor cannot trust the answer of a bad chip. Thus, the four possible outcomes of a test are as follows:

Chip A says	Chip B says	Conclusion
B is good	A is good	both are good, or both are bad
B is good	A is bad	at least one is bad
B is bad	A is good	at least one is bad
B is bad	A is bad	at least one is bad

   (a) Show that if more than n/2 chips are bad, the professor cannot necessarily determine which chips are good using any strategy based on this kind of pairwise test. Assume that the bad chips can conspire to fool the professor

   (b) Consider the problem of finding a single good chip from among n chips, assuming that more than n/2 of the chips are good. Show that [n/2] pairwise tests are sufficient to reduce the problem to one of nearly half the size.

   (c) Show that the good chips can be identified with (n) pairwise tests, assuming that more than n/2 of the chips are good. Give and solve the recurrence that describes the number of tests.

4. Use the master method to give tight asymptotic bounds for the following recurrences.
   (a) $T(n) = 2T(n/4) + 1$
   (b) $T(n) = 2T(n/4) + \sqrt{n}$
   (c) $T(n) = 2T(n/4) + n$
   (d) $T(n) = 2T(n/4) + n^2$
5. Professor Caesar wishes to develop a matrix -multiplication algorithm that is asymptotically faster than Strassen's algorithm . His algorithm will use the divide and conquer method, dividing each pieces of size $n/4 \times n/4$, and the divide and combine steps together will take $\Theta(n^2)$ time. He needs to determine how many sub-problems his algorithm has to create in order to beat Strassen's algorithm . if his algorithm creates a sub-problems, the the recurrence for the running time $T(n) = aT(n/4) + \Theta(n^2)$. What is the largest integer value of a for which professor Caesar's algorithm would be asymptotically faster than Strassen's algorithm ?
   Note : Strassen's algorithm takes $O(n^{2.71})$ time.