1. Solve each recurrence, expressing the answers using $O$, $\Omega$, or $\Theta$, whichever is most appropriate.

   (a) $F(n) = 4F(n/2) + n$

   (b) $F(n) = F(n/2) + \log n$ (Hint: use substitution.)

   (c) $F(n) = F(n - 2) + \log n$ (Hint: do not be misled by irrelevancies.)

   (d) $F(n) = F(n - \sqrt{n}) + n$ (Hint: divide by sides by something.)
(e) \[ F(n) = 3(F(n/3) + F(2n/3)) + n^2 \]

(f) \[ F(n) = F(n/2) + F(n/3) + F(n/6) + 1 \]

2. Explain how to find the median of \( n \) items, deterministically, in \( O(\log n) \) time using \( n \) processors. Can you do it with asymptotically fewer processors, but still in \( O(\log n) \) time?
3. Consider a union/find problem where there are $n$ items, and the total number of find operations is $n$ and the total number of union operations is also $n$. Assume that you use path compression.

(a) Is the time complexity $O(n)$? (Hint: No.)
(b) What is the time complexity?

4. 2$n$ items are placed into an open hash table of size $n$, using a pseudo-random hash function.

(a) What is the average number of items in a bucket? (Hint: 2.)
(b) Approximately how many buckets will have no items?
(c) Approximately how many buckets will have exactly one item?
(d) We say that a two items collide if they have the same hash value. Approximately how many other items does a given item $x$ collide with?
5. You are given an acyclic directed graph $G = (V, E)$.

(a) Write an algorithm which finds a topological ordering of $V$.

(b) Write an algorithm which finds the longest path in $G$. 
(c) Write an algorithm which finds the transitive closure of $G$.

(d) Write an algorithm which finds the transitive reduction of $G$. 
6. You can only type 80 characters on a line. You are given a sequence of words, $w_1, w_2, \ldots w_n$ of various lengths, which do not fit into one line. You want to construct a paragraph, where each line is as long as possible without exceeding 80 characters. The last line can have any length. No word has length greater than 80, and there must be a space between any two consecutive words. Design a dynamic programming algorithm for this problem. (There is a linear, that is, $O(n)$, time algorithm for this problem.)