Problem. 1 Order the following functions in increasing order of growth:

(a) $n!$  (b) $\log \log n$  (c) $n^{\log \log n}$  (d) $n^{1/\log n}$
(e) $2^{3 \log n}$  (f) $n^2$  (g) $(\log \log n)^n$  (h) $n \log^2 n$

Problem. 2 The input is a sequence of $n$ elements $x_1, x_2, \ldots, x_n$ that we can read sequentially. We want to use a memory that can only store $O(k)$ elements at a time. Give a high level description of an algorithm that finds the $k$th smallest element in $O(n)$ time.

Problem. 3 Let $L$ be a sequence of $n$ elements. If $x$ and $y$ are pointers into list $L$ then $\text{INSERT}(x)$ inserts a new element immediately to the right of $x$, $\text{DELETE}(x)$ deletes the element to which $x$ points and $\text{ORDER}(x, y)$ returns true if $x$ is before $y$ in the list. Show how to implement all three operations with worst case time $O(\log n)$.

Problem. 4 A simple undirected graph $G = (V, E)$ without self-loops has at most one edge between every pair of vertices and no edge from a vertex to itself. A graph is $p$-colorable if all vertices can be assigned one of $p$ colors with no edge receiving the same color at both of its ends.

Let $d(v)$ denote the degrees of a vertex $v$, i.e., the number of edges incident at $v$. Let $d(G)$ denote $\max_{v \in V} d(v)$, the maximum degree of the vertices of the graph $G$.

Design an efficient algorithm and prove its correctness, which determines $(d(G) + 1)$-coloring of the graph.