1 Heap
Do parts (a), (b) and (c) of Problem 6.2 (p.167) in textbook. For part (a), tell me where the $d$ children (if exists) of a node $i$ will be located in the array, and where the parent of a node $i$ will be in the array.

2 Lower bound
Consider the task of searching a sorted array $A[1 \ldots n]$ for a given element $x$: a task we usually perform by binary search in time $O(\log n)$. Show that any algorithm that accesses the array only via comparisons (that is, by asking questions of the form “is $A[i] \leq z$”), must take $\Omega(\log n)$ steps.

3 Counting Sort
Do Exercises 8.2-4 (on p. 197). In order to practice some implementaiton of the algorithm, you should write down the pseudo-code.

4 Coin Change
We now prove the simple greedy algorithm for the coin change problem with quarters, dimes, nickels and pennies are optimal (i.e. the number of coins in the given change is minimized), when the supply of each coin type is unlimited.

Let $q_o, d_o, k_o, p_o$ be the number of quarters, dimes, nickels and pennies used for changing $n$ cents in an optimal solution.

1. First, show that $d_o \leq 2, k_o \leq 1, p_o \leq 4$. Also, show that if $k = 1$, then $d \leq 1$.
2. Now show it is always optimal to choose as many quarters as possible.
3. Finally, show that the greedy algorithm is optimal in its choice for dimes, nickels and pennies.

5 Activity Selection
Let us consider an alternative greedy strategy for the activity selection problem. Let $\mathcal{R} = \{R_1, \ldots, R_n\}$ be the set of activities. Recall that each activity $R_i = [s_i, f_i]$.

1: while $\mathcal{R}$ is not empty do
2: Select the activity $R_i \in \mathcal{R}$ that overlaps with the fewest activities in $\mathcal{R}$ and add this activity to solution set $A$.
3: Remove $R_i$ and any activity $R_j$ that overlaps with $R_i$ from $\mathcal{R}$.
4: end while
Now prove or disprove that the greedy strategy solves the activity selection problem optimally.