Analyzing loops-10pts Consider the following algorithm, which takes as input two sorted (from smallest to largest) arrays $A$ and $B$ of size $n$ and for each $1 \leq I \leq n$ computes the minimum $1 \leq J \leq n$ for which $B[J] > A[I]$ and stores it in $C[I]$. (If no such $J$ exists, it sets $C[I]$ equal to $n + 1$.) Give a time analysis of the algorithm, up to $\Theta$.

SmallestLargerElement($A[1..n]$, $B[1..n]$)
1. $J \leftarrow 1$
2. FOR $I$=1 to $n$ do:
3. While $J \leq n$ and $B[J] \leq A[I]$ do $J +=$
4. $C[I] \leftarrow J$.

Greedy Algorithms and use of data structures in algorithms : 30 points total
Consider the following problem: You are making probes of an asteroid to reveal its chemical composition, by shooting lasers at certain frequencies at the asteroid. Each substance $s_i$ on a list of $n$ possible constituent substances has a range $[l_i, h_i]$ where it will react to the laser. You need to pick the smallest set of frequencies $f_1, .. f_k$ so that each $s_i$ will react to one of the frequencies $f_j$, i.e., for each $i$ there is at least one $j$ with $l_i \leq f_j \leq h_i$.

Part 1 : 5 points for counter-example Below is a greedy strategy for this problem that is not guaranteed to produce optimal solutions. Find an example where it fails to produce the optimal solution.
Candidate Strategy one: Pick a frequency $f$ that the maximum number of substances react to. Remove substances that react to $f$ and recurse, until all substances are covered.

Part 2: 5 pts Here is a greedy strategy that does produce optimal solutions. Illustrate the algorithm on the counter-example from Part 1.
Candidate Strategy two: Let $S_i$ be the substance with the smallest value of $h_i$. Pick frequency $f = h_i$. Remove substances that react to $f$, and recurse, until all substances are covered.

Part 2: 10 points For the optimal strategy, Candidate Strategy 2, prove that it is correct. You can use any valid proof method, but here are two possible outlines as hints:
Approach 1: Achieves the bound: Let $S_1, S_2, \ldots S_k$ be the substances chosen by the greedy strategy (i.e., $S_1$ has the smallest $h_i$, $S_2$ has the smallest $h_i$ once substances reacting to $h_1$ are removed, etc.) First prove that the corresponding intervals $[l_i, h_i]$ are disjoint. Use this to prove that any solution requires at least $k$ frequencies.

Approach 2: Greedy stays ahead: Let $\{f_1, \ldots f_t\}$ be the greedy set of frequencies, in increasing order, and let $\{f'_1, \ldots f'_t\}$ be any other set of frequencies, also in increasing order. Prove by induction that $f_i \geq f'_i$. Then use this to show that $t \leq t'$.

Part 4: 10 points For the optimal strategy (Strategy 2), describe an efficient algorithm that carries out the strategy. Your description should specify which data structures you use, and any pre-processing steps. Give a time analysis.