Question #1:
You are given an array $A$ of $n$ positive integer numbers. All the numbers in $A$ are in the range $[10, 10n]$ except for 10 numbers. Can you design an algorithm to sort the array $A$ in $O(n)$ time?

Question #2:
Let $I_1, \ldots, I_n$ be $n$ intervals, where interval $I_i$ is defined by set $[a_i, b_i]$, i.e., starting from $a_i$ and ending at $b_i$. The interval scheduling problem asks to find a maximum number of disjoint intervals that do not overlap with each other (for example, if four intervals are given as $I_1 = [1, 2], I_2 = [2, 3], I_3 = [1, 4], I_4 = [4, 5]$, then the solution is $\{I_1, I_2, I_4\}$). Can you design a linear time greedy algorithm to solve the interval scheduling problem? You can ignore the running time of your algorithm in the pre-processing phase.

Question #3:
The input consists of a directed graph, encoded in the usual adjacency list representation, along with two distinguished vertices $s$ and $t$. Design and analyze a linear time algorithm to decide whether or not there is a path from $s$ to $t$ that contains a cycle.