1. (18 points) We are given two sets $A$ and $B$ with $n$ elements each in the form of arrays. We are also given that $A$ is in sorted order and $B$ may not be in sorted order. In addition, $|A \cap B| = \sqrt{n}$. Present an $\tilde{O}(\sqrt{n} \log^2 n)$ time Las Vegas algorithm to output an element that is common to $A$ and $B$. 
2. (16 points) A department has to keep records of its employees such that the following operations can be performed on the records:

- **Find_Name(SSN):** Return the name of the person whose social security number is SSN; and
- **Find_SSN(Name):** Return the social security number of the person whose name is Name.

Present a data structure for keeping the records that will take $O(\log n)$ time to perform each of the above operations, $n$ being the number of persons in the department. You can use $O(n)$ space.
3. (17 points) Present a data structure that can support the following operations:

  - **INSERT**\( (x) \) – Insert the element \( x \) into the data structure if \( x \) is not already there.
  - **FIND\_MIN()** – Return the value of the smallest element in the data structure.
  - **DEL\_MIN()** – Delete and return the minimum value from the data structure.
  - **INC\_ALL\( (y) \)** – Increase the value of each element in the data structure by \( y \).

Each operation should take \( O(\log n) \) time and you can use \( O(n) \) space.
4. (16 points) Two different divide-and-conquer algorithms $A$ and $B$ have been designed for solving the problem $\pi$. $A$ partitions $\pi$ into 5 subproblems each of size $\frac{n}{3}$. Here $n$ is the input size for $\pi$. It takes a total of $\Theta(n^2)$ time for the partition and combine steps. $B$ partitions $\pi$ into 10 subproblems each of size $\frac{n}{4}$. It takes a total of $\Theta(n^{1.8})$ time for the partition and combine steps. Which algorithm is preferable? Why?
5. (16 points) Input are an array $a[\ ]$ of $n$ elements and an element $x$. We have to check if there are three elements in $a[\ ]$ whose sum is $x$. Assume that the array elements are distinct. Present an $O(n^2 \log n)$ time algorithm for this problem.
6. (17 points) Input are $k$ sets $S_1, S_2, \ldots, S_k$ such that $\sum_{i=1}^{k} |S_i| = n$. The elements of these sets are integers in the range $[1, n^{10}]$. The problem is to sort these $k$ sets. Present an algorithm to sort all of these sets in a total of $O(n)$ time.