1. Find an efficient data structure for representing a subset $S$ of the integers from 1 to $n$. Operations we wish to perform on the set are:
   - **INSERT**(i): insert the integer $i$ to the set $S$. If $i$ is already in the set, this instruction must be ignored.
   - **DELETE**: delete an arbitrary member from the set.
   - **MEMBER**(i): check whether $i$ is a member of the set.

   Your data structure should enable each one of the above operations in constant time (irrespective of the cardinality of $S$).

2. Input is a sequence $X$ of $n$ keys with many duplications such that the number of distinct keys is $d(< n)$. Present an $O(n \log d)$-time sorting algorithm for this input. (For example, if $X = 5, 6, 1, 18, 6, 4, 4, 1, 5, 17$, the number of distinct keys in $X$ is six.)

3. Solve the following recurrence relations:
   
   (a) \[
   T(n) = \begin{cases} 
   1 & \text{if } n \leq 3 \\
   16T(n/3) + n^2 & \text{if } n > 3 
   \end{cases}
   \]

   (b) \[
   T(n) = \begin{cases} 
   1 & \text{if } n \leq 4 \\
   T(\sqrt{n}) + \log n & \text{if } n > 4 
   \end{cases}
   \]

4. Given are two sets $A$ and $B$ with $m$ and $n$ elements, respectively, from a linear order. These sets are not necessarily sorted. Also assume that $m \leq n$. Show how to compute $A \cup B$ and $A \cap B$ in $O(n \log m)$ time.

5. $X_1, X_2, \ldots, X_\ell$ are sorted sequences such that $\sum_{i=1}^\ell |X_i| = n$. Show how to merge these $\ell$ sequences in $O(n \log \ell)$ time.