1. In this problem we consider a data structure for maintaining a multi-set $M$. We want to support the following operations:

- $Init(M)$: create an empty data structure $M$.
- $Insert(M, i)$: insert (one copy of) $i$ in $M$.
- $Remove(M, i)$: remove (one copy of) $i$ from $M$.
- $Frequency(M, i)$: return the number of copies of $i$ in $M$.
- $Select(M, k)$: return the $k$’th element in the sorted order of elements in $M$.

If for example $M$ consists of the elements $<0, 3, 3, 4, 4, 7, 8, 8, 9, 11, 11, 11, 11, 13>$, then $Frequency(M, 4)$ will return 2 and $Select(M, 6)$ will return 7.

Let $|M|$ and $\|M\|$ denote the number of elements and the number of different elements in $M$, respectively.

a) Describe an implementation of the data structure such that $Init(M)$ takes $O(1)$ time and all other operations take $O(\log \|M\|)$ time.

b) Design an algorithm for sorting a list $L$ in $O(|L| \log \|L\|)$ time using this data structure.

2. The mean $M$ of a set of $k$ integers $\{x_1, x_2, \ldots x_k\}$ is defined as

$$M = \frac{1}{k} \sum_{i=1}^{k} x_i.$$

We want to maintain a data structure $D$ on a set of integers under the normal $Init$, $Insert$, $Delete$, and $Find$ operations, as well as a $Mean$ operation, defined as follows:

- $Init(D)$: Create an empty structure $D$. 
**INSERT(\(D, x\))**: Insert \(x\) in \(D\).

**DELETE(\(D, x\))**: Delete \(x\) from \(D\).

**FIND(\(D, x\))**: Return pointer to \(x\) in \(D\).

**MEAN(\(D, a, b\))**: Return the mean of the set consisting of elements \(x\) in \(D\) with \(a \leq x \leq b\).

(a) What does MEAN(\(D, 7, 17\)) return if \(D\) contains integers 
\(\{2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 27\}\)?

(b) Describe how to modify a standard red-black tree in order to implement \(D\) such that INIT is supported in \(O(1)\) time and INSERT, DELETE, FIND, and MEAN are supported in \(O(\log n)\) time.