Suppose a bank has a collection of accounts, each identified with a (distinct) owner name, and each having a value, namely the balance. The bank wishes to support the following operations:

1. Create an account.
2. Close an account.
3. Add (or subtract) a given sum from the balance for an account identified by the owner’s name.
4. Report the account with maximum balance.
5. Report the balance to an owner on request.

Show how to support these operations so they run in time $\Theta(\log n)$, where $n$ is the number of accounts.

6. Consider maintaining a database for a biologist storing results concerning pea plant experiments. Each experiment records a distinct (integer) ID, a plant weight and a plant height. We would like to support the following query:

   For plants with heights in the range $[h_1, h_2]$, what is the average weight?

   Also insertions and deletions of experiments will be allowed.

   Show how to support these operations so they run in time $\Theta(\log n)$, where $n$ is the number of experiments recorded in the database.

7. Suppose that we are maintaining a set of items with integer keys. In addition to their key field, items have a second attribute, called their color, which can take on the values red (R) or green (G). We want to support the following four operations:

   • Insert item $e$.
   • Delete item $e$.
   • Report the color of item $e$.
   • The operation $\text{Flip}(k_1, k_2)$. It switches the color of all items with key value between $k_1$ and $k_2$ inclusive.

   Show how to implement these operations so that they all run in $O(\log n)$ time.

   Hint. Use Boolean bits on the edges to indicate whether to flip the colors of items in the subtree at the bottom of the edge.