Problem 4.1 [Dynamic Dictionary with Working-Set Property].

A binary search tree has the \textit{worst-case working-set property} if every access $x_i$ costs $O(\log t_i)$ worst-case time, where $t_i$ is the number of distinct keys accessed since the last access to key $x_i$.

Describe and analyze a dynamic dictionary (not necessarily a BST) that has the working-set property. Your data structure should:

(a) use $O(n)$ space, where $n$ is the current number of items in the dictionary;

(b) support searching for key $x_i$ in $O(\log t_i)$ worst-case time, where $t_i$ is the number of distinct keys accessed since the insertion or last access to the key $x_i$; and

(c) support insertions and deletions in $O(\log n)$ amortized time.

\textit{Hint:} Consider representing your dictionary as a list of binary search trees of increasing size.