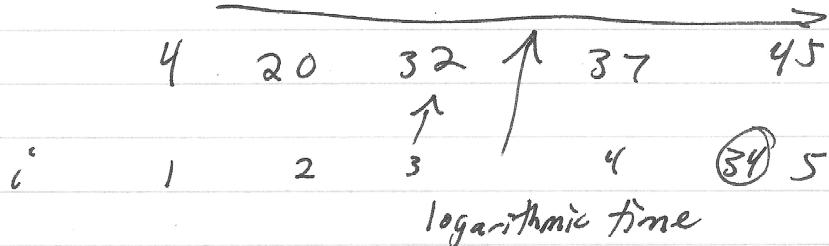


Unsorted list/array : needs to run in $O(1gn)$ and this runs usually in $O(n)$
 \Rightarrow Insert $O(1)$
 \Rightarrow Cheat $O(n)$

Sorted list/array



$O(\lg n)$ search

$O(1)$ to check

Insertion: have to shift everything, thus adding to set R is $> O(\lg n)$

Find smallest i such that $R[i] \geq t$ in $O(\lg n)$ time
(compare $R[i]$ and $R[i-1]$ against t in $O(1)$ time.)

Actual insertion requires shifting $\rightarrow O(n)$ time

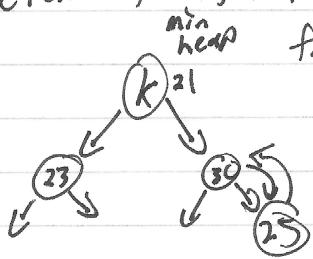
Sorted list: $\square \rightarrow \square \rightarrow \square \rightarrow \square$

 ↑ insert point

move pointers in $O(1)$ but can't use binary search

Heaps: min/max

element that is $\leq k$ or $\geq k$
from t is $O(n)$ time



insert 25

\Rightarrow only work on right side of heap, so inserting into set R requires checking both sides and is $O(n)$ time