Hash Tables

hash code function h : Keys $\longrightarrow$ Hashes (int / long)

(many to 1)

① given (key, value) pair, i.e. (k, v) — find $h(k)$

($k \to h(k)$, $k' \to h(k)$ possible)

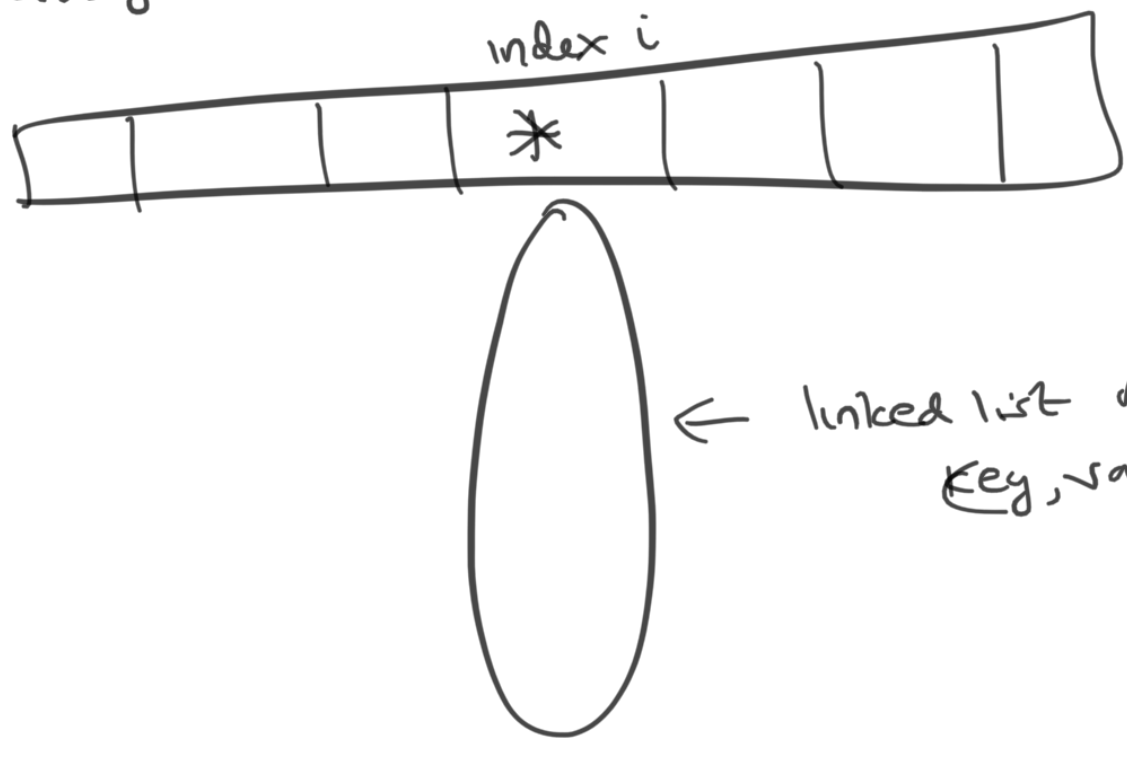
② map $h(k)$ to index in array

(eg) $h(k)$ % array_length

= index of $h(k)$ in array

$h(k) \neq h(k'')$ → same index in array possible

③ array



Retrieve

find $d(k)$ .... ,

① find $h(k)$

② find i = index corresponding to $h(k)$

③ in arr[i] ← linked list, search for k and retrieve its value

# of collisions high ⇒ $O(N)$ look up runtime

N = # of keys

# of collisions low/min ⇒ $O(1)$ look up runtime

(linked list at each arr[i] ≃ constant length)

another option: → instead of arrays, used balanced binary search tree.. $O(\log N)$ look up time