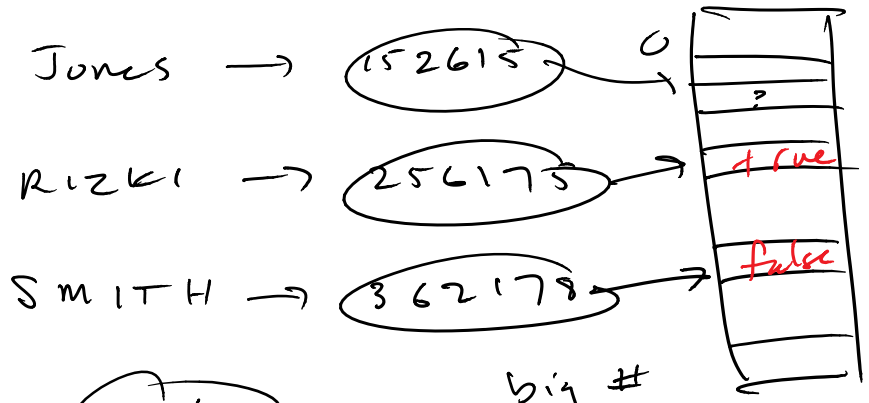
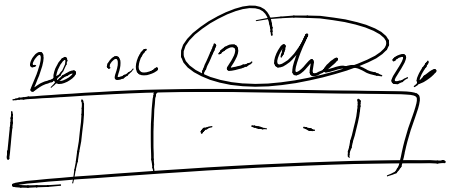
		Insert/Remove	Search
Array	sort	O(N)	O(logn)
	unsort	O(1)	O(N)
List		O(1)	O(N)
Tree	balanced	O(logn)	O(logn)
		O(1)	O(1)



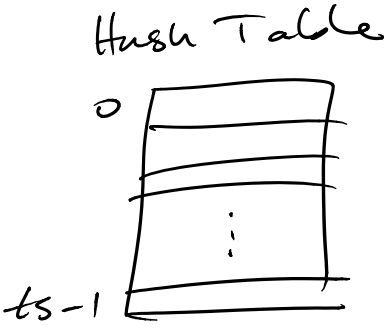


h: Key Space → address space
very large moderate

Hash Table

Hash function

$h(key)$ = table location



Keys	h(keys)
27	5
15	4
18	7
26	4
14	3

collision

collision handling technique

O(ts)

$h(key) = key \% ts$

$= key \% 11$

linear probing to resolve collision

$h(key) = (key + i) \% ts$ $\quad i = 1, 2, 3 \ldots ts-1$

probability of a collision

$\lambda = \frac{N}{ts} = \frac{5}{11}$

$\lambda = \frac{N}{ts} = \frac{5}{11}$

load factor

$ts = 11$

---

linear probing $h(k) = (k + i) \% ts$ $i = 0, 1, 2, 3 \ldots$

quadratic probing $h(k) = (k + i^2) \% ts$ $i = 1, 2, 3 \ldots$

---

double hash

$h(k) = k \% ts$

$h'(k) = k \% ts'$

$(h(k) + (h'(k) \% ts) \cdot i) \% ts$

$$h(k) = (k + (h'(k) \% ts) \cdot i) \% ts$$

$i = 1, 2, 3 \ldots$

$h(k) = k \% 11$

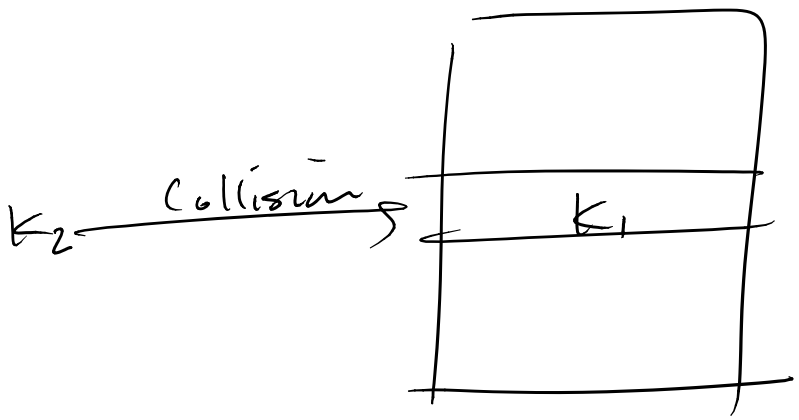
$h'(k) = (k \% 7) + 1$

$h(k) = (k + h'(k) * i) \% 11$

Keys	h(k)
27	3
15	4
16	7
14	3 → 4 → 5
26	4 → 10

$(14 \% 7) + 1 = 1$

$(26 \% 7) + 1 = 6$



linear tests every where

Quadratic if table is prime#

double hash ts prime#

double room
ts prime#
ts' prime#