Date) 28/02/23)

{Data Structures}

Hashing :: It also a Searching technique.

What is Set?

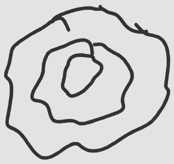
It is a Collection of objects themself are Called Elements.

for Exa = A Set of Natural numbers $N = \{1, 2, \ldots N\}$

$A = \{0, 2, 4, 6, 8\}$ Set of Non-negative Even numbers Less than 10.

Elements of Set.

$A = \{11, 12, 13, \ldots\}$ Set of all numbers greater than 10.



Empty Set

$A = \{1, 2, 3, 4, 5, 6\}$

$B = \{2, 4, 6\}$

$A \cup B = \{1, 2, 3, 4, 5, 6\}$

$A \cap B = \{2, 4, 6\}$

$A - B = \{1, 3, 5\}$

$B - A = \{\emptyset\} \quad \{\ \} \quad \emptyset$

The bit string of the sets $\{1, 2, 3, 4, 5\}$ and $\{1, 3, 5, 7, 9\}$ are 1111100000 and 1010101010 respectively. Use bit string find the union and intersection of sets.

$A = \{1, 2, 3, 4, 5\} = 1111100000$

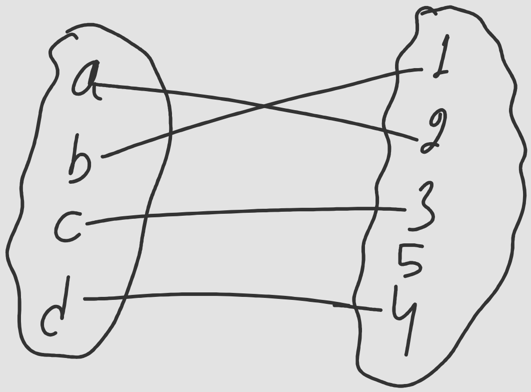
$B = \{1, 3, 5, 7, 9\} = 1010101010$

$A \cap B = \{ \quad \} = 1010100000$ ✓

$A \cup B = \{ \quad \} = 1111101010$ ✓

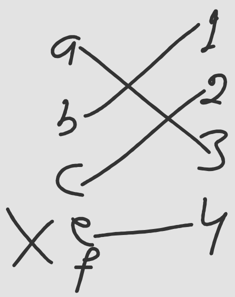
one-one & onto

$f(x_1) = f(x_2)$ iff $x_1 = x_2$

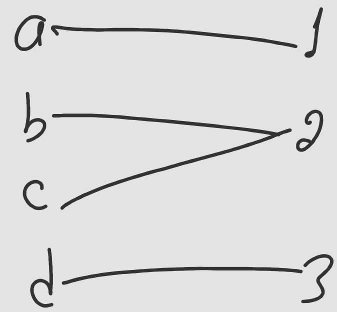


one-one

$f(x) = x^2$ is it one-one.



one-one.
not onto



onto

$y = x$

multisets::

$A = \{1, 1, 1, 2, 2, 3\}$

$S = \{3.1, 2.2, 1.3\}$

n_1 = number of elements.

element

$S = \{n_1.a_1, n_2.a_2, n_3.a_3, \dots n_i.a_i\}$

$A = \{1, 1, 1, 2, 2, 3\}$

$B = \{1, 1, 4, 3\}$

$A \cup B = \{1, 1, 1, 2, 2, 3, 4\}$

$A \cap B = \{1, 1, 3\}$

$A + B = \{1, 1, 1, 1, 1, 2, 2, 3, 3, 4\}$

$A - B = \{1, 2, 2\}$

$A = \{3.a,\ 2.b,\ 1.c\}$

$B = \{2.a,\ 3.b,\ 4.d\}$

$A \cup B = \{3.a,\ 3.b,\ 1.c,\ 4.d\}$

$A \cap B = \{2.a,\ 2b\}$

$A - B = \{1.a,\ 1.c\}$

$B - A = \{1b,\ 4.d\}$

Hashing technique:: It is a Searching method

Linear Search::

A =

8	3	5	7	2	4	6	1
0	1	2	3	4	5	6	7

$O(n)$

key = 2

Key = 9

$O(1)$, $O(1)$

Best Case   Worst Case

Binary Search

B =

2	4	6	7	9	12	15	20
0	1	2	3	4	5	6	7

Key = 12

9 ~~12~~

12 > 9

$T(n) = \underline{T(n/2)} + O(1)/c/1$

0, 1, 2, 3, 4, 5, 6, 7, 8

$O(1)$ $mid = \frac{8+0}{2} = 4$

If(Key == A[mid]) $O(1)$

return mid.

$T(n) = T(n/2) + 1$

$O(\lg n)$

$T(n) = T(n/2) + 1$

$T(n/2) = T(n/4) + 1$

$T(n/4) = T(n/8) + 1$

$\rightarrow O(1) \rightarrow c$

c

$c + c + c + \dots$

c

c

$1 - n$

log(n) times

$\frac{n}{2^k} = 1$

$n = 2^K$

$K = \log_2 n$

$T(n) = 1 + K$

$= 1 + \log_2 n$

$T(n) = O(\log_2 n)$

32, 16, 8, 4, 2, 1

$T(n) = T(n/2) + 1 \Rightarrow T(n) = T(n/8) + 3$

$T(n/2) = T(n/4) + 1 \Rightarrow T(n/2) = T(n/8) + 2$

$T(n/4) = T(n/8) + 1$

$\vdots$

After 3rd iteration

$T(n) = T(n/8) + 3$

$T(n) = T(n/2^3) + 3$

After k^{th} iteration

$T(n) = T(n/2^k) + k$

$T(n) = \underline{T(1)} + k$

$n = 2$'s power

$n = n/2^0$

$n/2 = n/2^1$

$n/4 = n/2^2$

$\vdots$

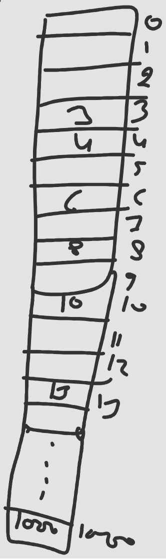
$n/2^k = 1$

1

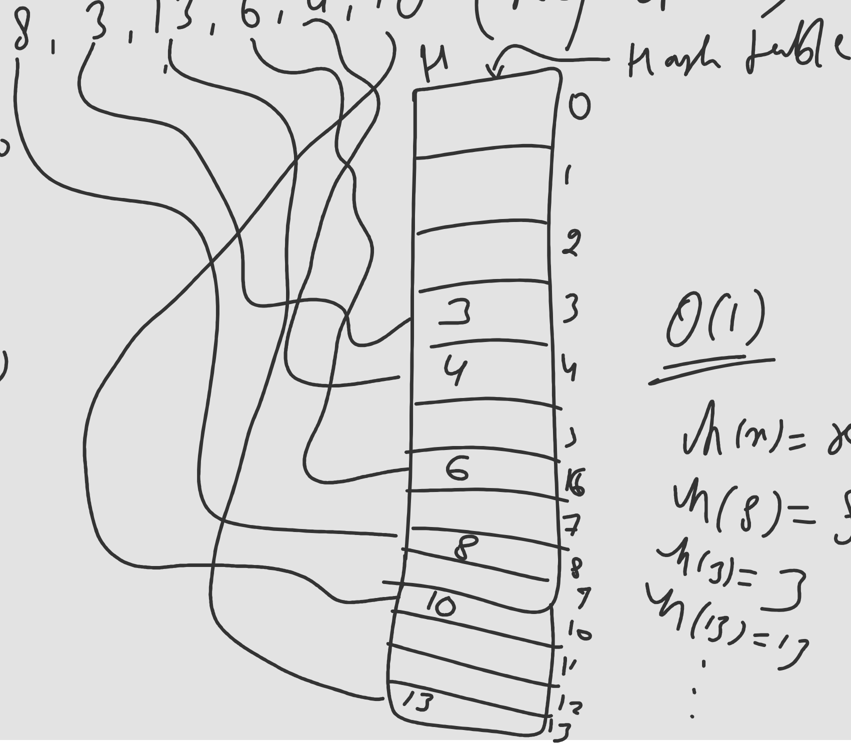
Hashing :: Hashing is a technique or process of mapping keys and values into hash table using hash function.

Given Keys :: 8, 3, 13, 6, 4, 10 (Key Space)

8, 3, 13, 6, 4, 10, 1000



$O(1)$



$\underline{O(1)}$

$h(x) = x$

$h(8) = 8$

$h(3) = 3$

$h(13) = 13$

$\vdots$

$O(n)$

$\theta(n)$

$E(T(n))$

-18

Modified hash function:

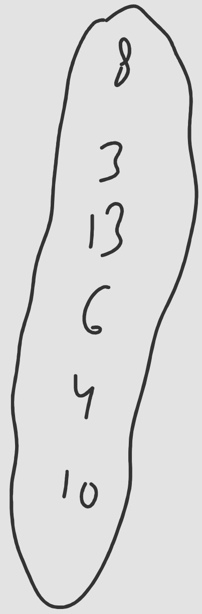
$h(x) = x \% \text{size}.$

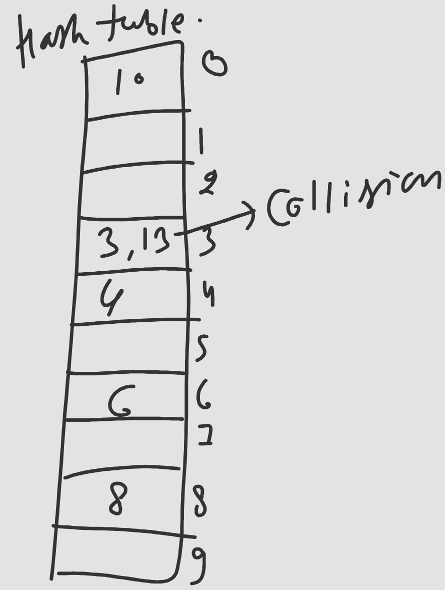
$h(8) = 8 \% 10 = 8$

$h(3) = 3 \% 10 = 3$

$h(13) = 13 \% 10 = 3$

$h(6) = 6 \% 6 = 6$

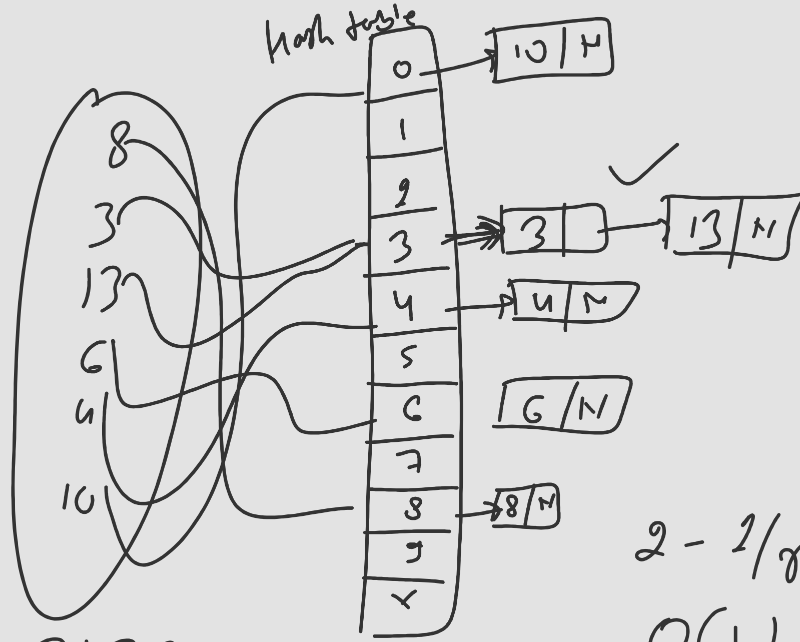




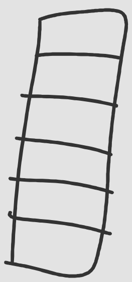
Collision resolution methods.

1) open hashing ✓
   - chaining ✓

2) closed hashing
   i) Linear Probing
   ii) Quadratic Probing



key = 13



71, 72, 73, 74, 71, 72, 73, ...

$2 - 1/n$

$O(1)$

Worst Case = $O(n)$

Closed hashing
i) Linear probing
ii) Quadratic probing

i) Linear probing :: $h(x) = x \,\%\, \text{size}$ ✓ ← $\underline{\underline{10}}$

$h'(x) = [h(x) + f(i)] \,\%\, \text{size}$

$f(i)$ — linear function

$h'(8) = [8 \% 10 + 0] \% 10 = 8$

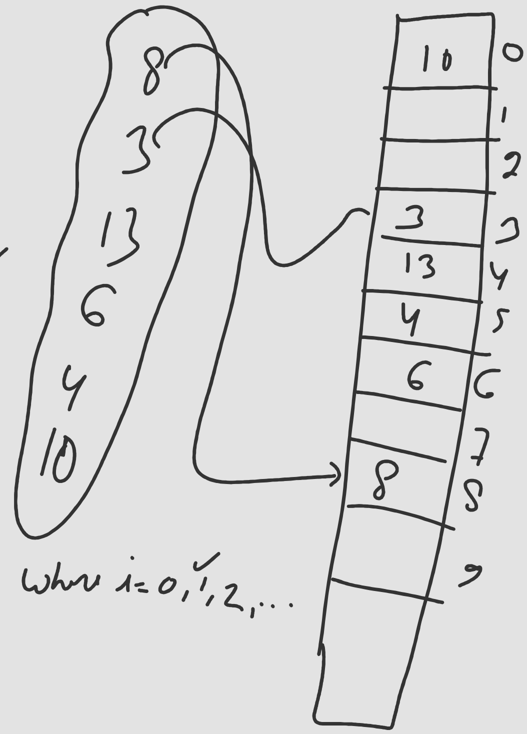
$h'(3) = [3 \% 10 + 0] \% 10 = 3$

$h'(13) = [13 \% 10 + 0] \% 10 = 3$

$h'(13) = [13 \% 10 + 1] \% = 4$

$f(i) = i$ where $i = 0, 1, 2, \ldots$

$h'(4) = 4$



§ Quadratic probing::

$h'(x) = [h(x) + f(i)] \% \text{size}$

$f(i) = i^2$

$h'(8) = [8\%10 + 0]\%10 = 8$

$h'(3) = [3\%10 + 0]\%10 = 3$

$h'(13) = [13\%10 + 0]\%10 = 3$

$h'(13) = [13\%10 + 1]\%10 = 4$

$f(i) = i^2$

$h'(23) = [23\%10 + 0]\%10 = 3$

$h'(23) = [23\%10 + 1]\%10 = 4$

$h'(23) = [23\%10 + 4]\%10 = 7$

$h'(43) = [43\%10 + 9]\%10 = 2$