Hashing

It is the process of mapping the large amount of data item to a smaller table with the help of hash function. ie., hashings lets us create a map where each string (value) is mapped to a unique key by a function known as hashing function

If H denotes the hash function and x is the data, then hash key 'y' can be generated as $y = H(x)$;

If x_1 and x_2 are two datas, we should never have a scenario where in $H(x_1) = H(x_2)$

Almost all hash functions in use are imperfect that means, we might end up in a situation wherein $H(x_1) = H(x_2)$, such a condition is termed as 'collisio

ex:- $H(x) = x \mod 10$

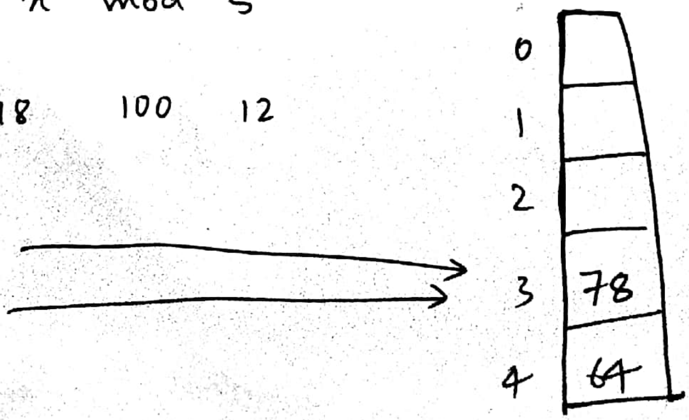
Eg:-

$H(x) = x \mod 5$

64 78 98 100 12

$H(78) = 3$

$H(98) = 3$



Collision in hashing can be avoided using the following methods,

hash table

i) Open Adressing

ii) Closed Adressing. (seperate chaining.)

open — linear probing
— quadratic "
— double hashing.

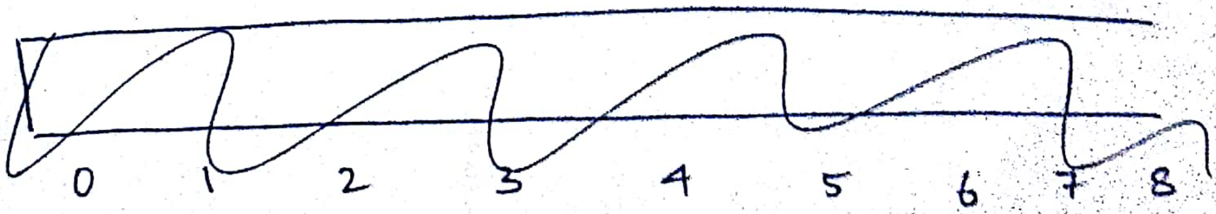
Open Addressing

In open addressing, when a new key collides, find the next empty slot and store the data there.
while searching for the next empty slot, if in case we reach end of table., start searching from the beginning of the table.

∴

0	98
1	100
2	12
3	78
4	64

Q) For the input, 0 1 4 9 16 25 36 49 64 81 100 and H(x) = x mod 7, construct open hash table.

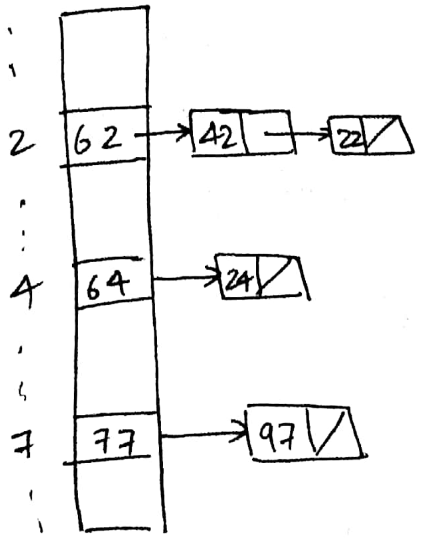


Closed Addressing (seperate chaining)

all the collided items are added at the end of the list whose header is plugged at the hashed location in the hash table.

62 64 24 77 42 97 22

$H(x) = x \bmod 10.$


2
62
42
22
4
64
24
7
77
97