

"HASHING"

DEFINITION:-

> Hashing is an effective way to store the elements in some data structure.

> It allows to reduce the number of comparisons.

> Using the hashing technique we can obtain the concept of direct access of stored record.

> There are two concepts used regarding the hashing. They are,

→ Hash table

→ Hash function.

Hash Table:-

> Hash table is a data structure used for storing and retrieving data very quickly.

> Insertion of data in the hash table is based on the key value.

> Hence every entry in the hash table is associated with some key.

> For example for storing an employee record in the hash table the employee ID will work as a key.

Hash Function:-

- > Hash Function is a function which is used to put the data in the hash table.
- > Hence one can use the same hash function to receive the data from the hash table.
- > Thus hash function is used to implement the hash table.
- > The integer returned by the hash function is called hash key.

Types of hash functions:-

> There are various types of hash functions that are used to place the record in the hash table.

> They are namely given below,

1. Division Method:-

> The hash function depends upon the remainder of division. Typically the divisor is the table length.

Ex:-

> if the record 54, 72, 89, 37 is to be placed in the hash table and if the table size is 10. Then,

$$h(\text{key}) = \text{record} \% \text{table size}$$

$$4 = 54 \% 10$$

$$2 = 72 \% 10$$

$$9 = 89 \% 10$$

$$7 = 37 \% 10$$

0	
1	
2	72
3	
4	54
5	
6	
7	37
8	
9	89

2) MID SQUARE:-

> In the mid square method, the key is squared and the middle (or) mid part of the result is used as the index.

> If the key is a string, it has to be preprocessed to produce a number.

Example:-

> consider that if we want to place a record 3111 then,

$$3111^2 = 96783216$$

For the hash table of size 1000

$$H(3111) = 783$$

3) Multiplicative hash function:-

> The given record is multiplied by some constant value. The formula for computing the hash key is.

$$H(\text{key}) = \text{floor} (P * (\text{fractional part of key} * A))$$

Where,

P = integer constant

A = constant Real Number

> Donald Knuth suggested to use constant

$$A = 0.618033.$$

Example:-

if $\text{key} = 107$ & $P = 50$ Then

$$H(\text{key}) = \text{floor} (50 * (107 * 0.618033))$$

$$= \text{floor} (3306.48184)$$

$$H(\text{key}) = 3306$$

> At 3306 location in the hash table the record 107 will be placed.

4) Digit folding:-

> The key is divided in to separate parts and using some simple operation these parts are combined to produce the hash key.

Example:-

> consider the record 12365412 then it is divided into separate parts as 123 654 12 and these are added together,

$$\therefore H(\text{Key}) = 123 + 654 + 12 \\ = 789.$$

> The record will be placed at location 789 in the hash table.

COLLISION:-

> The situation in which the hash function returns the same hash key for more than one record is called "collision". and two same hash key returned for different records is called synonym.

COLLISION RESOLUTION STRATEGIES:-

> If collision occurs then it should be handled by applying some techniques. Such a technique is called collision handling technique.

> Some of the collision Resolution strategies are given below,

a) Separate chaining

b) Open addressing.

a) SEPARATE CHAINING:-

> In collision handling method chaining is a concept which introduces an additional field with data i.e. chain.

> A separate chain table is maintained for colliding data.

> If collision occurs then a linked list (chain) is maintained at the home bucket.

Example:-

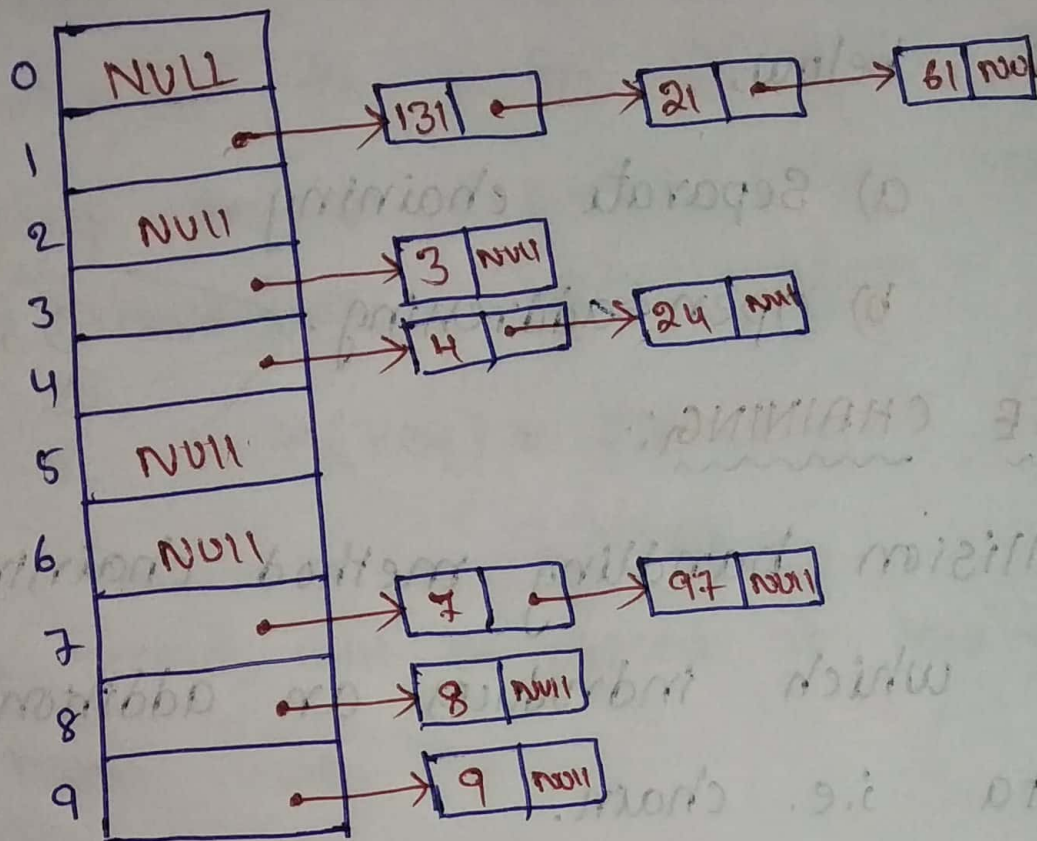
> consider the keys to be placed in their home buckets are,

131, 3, 4, 21, 61, 24, 7, 97, 8, 9

Then we will apply a hash function as

$$H(\text{key}) = \text{key} \% D$$

Where $D = \text{Table size}$. i.e. $= 10$.



→ open addressing

→ linear probing

→ quadratic

→ double hashing