Date:- / /
Page No.:- 1

Title :- Telephone database using Hash Table

Problem Statement :- consider telephone book database of N clients. Make use of a hash table implementation to quickly look up client's telephone number. make use of two collision handling techniques and compare them using number of comparisions required to find a set of telephone numbers.

Objectives :- To understand the ~~Concept~~ concept of Hashing and to understand to find record quickly using hash functions.

Theory :- Hash table is an efficient implementation of keyed array data structure. a structure sometimes known as a association array or map. Hash Table is a data structure. which each data value has its own unique index value. Access of data becomes very fast if we know the index of the desired data.

Thus, it is a data structure in which insertion and search operations are very fast irrespective of the size of the data Hash table uses an array as a storage medium and uses hash technique to generate an index where an element is to be inserted or is to be located from.

- Hashing
  - Hashing is a technique to convert a range of key values into a range of an array. we are going to use modulo operator to get a range of key values.

Example, Keys ⇒ 3, 7, 15, 9, 2, 10, 21, 90

index	Keys.
0	90
1	21
2	2
3	3
4	-
5	15
6	16
7	7
8	-
9	9

Hash Table.

$90 \% 10 = 0$

$21 \% 10 = 1$

$2 \% 10 = 2$

$3 \% 10 = 3$

$15 \% 10 = 5$

$16 \% 10 = 6$

$7 \% 10 = 7$

$9 \% 10 = 9$

- linear probing
  - As we can see, it may happen that the hashing technique is used to create an already used index of array. In such a case, we can search the next empty location in the array be looking into the next cell until we find an empty cell. This technique is called as linear probing

Example Keys ⇒ 1, 2, 42, 4, 12, 14, 17, 13, 37

Index	Keys	
0		
1	1	1 % 10 = 1
2	2	2 % 10 = 2
3	42	42 % 10 = 2 collision, 2+1 = 3 probing
4	4	4 % 10 = 4
5	12	12 % 10 = 2 collision, 2+1 = 3+1 = 4+1 = 5
6	14	14 % 10 = 4 collision, 4+1 = 5+1 = 6
7	17	17 % 10 = 7
8	13	13 % 10 = 3 collision 3+1 = 4+1 = 5+1 = 6 6+1 = 7+1 = 8.
9	37	37 % 10 = 7 collision 7+1 = 8+1 = 9.

- Following are the basic primary operations of a hash table.
  1] search - Searches an element in a hash table.
  2] Insert - inserts an element in a hash table.
  3] Delete - Deletes an element from hash table.

- Algorithm
  1. Start
  2. Choose a hash function
  3. choose an empty hash table.
  4. Insert a new client
  5. Look up a client's telephone number
  6. a. calculate the hash value, h, for the client's name. using the same hash function as before.
     b. check if the slot at index h is empty. if it is, the client is not in the database.
     c. if the slot at index h is occupied, traverse the linked list until the client with the matching name is found.

Date:- / /
Page No.:- 4

c. if the slot at index h is occupied, traverse the linked list until the client with the matching name is found. If the name is found, the client is in the database.

d. once the maching client found, return their telephone number.

6. STOP.

Conclusion:- Using a hash function implementation to quickly look up a clients telephone number is an efficient and effective approach for a telephone book database of N. clients.