* Assignment No:8 *

*Title: Implement all function of n dictonry (ADT). Using hash function.

* Objective:
1) To Understand dictonary (ADT).
2) To Understand concept of hashing.
3) To Understand concept of features like Searching.

* Learning Objective:
1) To Understand dictonary (ADT).
2) To Understand concept of hashing.

* problem statement:-
Implement all the function of a dictonary (ADT) Using hashing
Data:- Set of (key, value) pairs, keys are mapped to values, keys must to comparable, keys must be unique.

*Standered operation: Insert (key, value), find(key). Delete(key).

* Learning outcomes:
i) Define class dictonaries using group.
ii) Aonlyze Undeostood concept of hasning

Theory:-

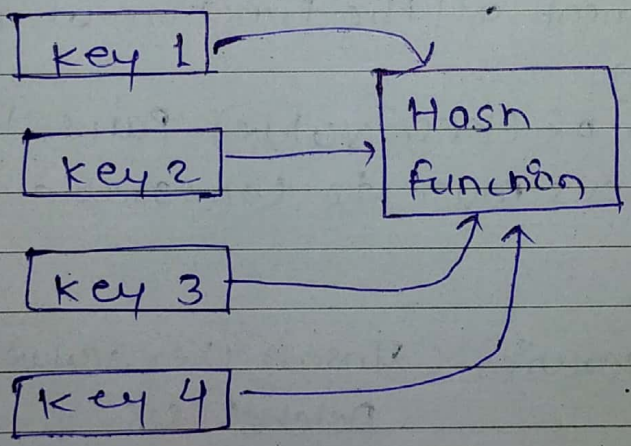
- Dictionary ADT:-

Dictionary is Data structure which is generic on assaciation of unique keys, with some value the may.

bind the value to key values are not required to the unique.

* Hashing:-

Hashing is a technique be convert a range of key value into indexes of an array no get a range.



index	value
0	value-1
1	value-2
2	value-3
3	value-4
4	value-5

* Basic Operation of hash table:-

- Search :- Searches an element in a hash table.
- Insert :- Insert an element in hash table.

delete: delete an element from hash table.

- Data item:

Define a data item having some Data and key based on which the search is to be occluded in hash table.

```
Struct Data_item
{
    int data;
    int key;
};
```

- Dictionary Operation:

(1) Dictionary create()-:
create empty dictionary

(2) put (dictionary d, key k, value v)
associate key k with values v if key k also already present in the dictionary old value of replaced by v.

(3) Value get (dictionary d, key k)
return a value associated with key or null if dictionary contain no such key.

(4) Remove: Remove key k, and associate value.

(5) Destory (dictionary d)
destory dictionary d;

Hash table is a data structure which stores data in an associative manner. In a hash table data is stored in array format where each data value has its own unique index value Access of data becomes very fast if we know the index of desired data.

Hash Method :-

Define a hashing method to compute the hash code of the key of data item.

```
int hashcode (int key)
{
    return key % Size;
}
```

Search Operation :

Whenever an element to be searched compare the hash code of key passed & locate the element using that hash function code as index in array.

- Insert operation :

Whenever an element is to be inserted compute the table code of key passed and locate the using that the code on index in array also linear probing.

for. e.g.:

• word:

① Limpid.
② Significator
③ Taximeter
④ Brightly
⑤ Scattering
⑥ Fieldstone.
⑦ Oxfordshire.

Hash table: 25

H(L) = 76 % 25 = 1
H(S) = 83 % 25 = 8.
H(T) = 84 (%) 25 = 9
H(B) = 66 % 25 = 16.
H(S) = 83 % 25 = 8
H(F) = 70 % 25 = 8 collision.
H(O) = 79 % 25 = 4

	Word.	Meaning.
0		
1	Limpid.	-
2		
3		
4		
5		
6		
7		
8	Significator	-
9	Taximeter.	-
10		
11		
12		
13		
14		
15		
16	Brightly	-
.	Scattering.	-
20	fieldstone..	-
25		

oflowchart.!-

Start

Input key &
value of dictionary

calculate the addr.
Using hash fun^n.

if
hash[add]
= -11

Hash[add]:key
flag:1

Hash[add+1]
hash[add]=key
flag=1

if
flag=0.

Search empty
location fun^n of index
to add index

if found assign
key that addre.

Stop.

o Conclusion :-

Hence we studied and implemented the dictionary (ADT) hashing.