**[Hashing in Data Structure | Hash Functions](https://www.gatevidyalay.com/hashing/)**

## **Searching Techniques**

* There are several searching techniques like linear search, binary search, search trees etc.
* In these techniques, time taken to search any particular element depends on the total number of elements.

### **Example-**

* [**Linear Search**](https://www.gatevidyalay.com/linear-search-searching-algorithms/) takes O(n) time to perform the search in unsorted arrays consisting of n elements.
* [**Binary Search**](https://www.gatevidyalay.com/binary-search-binary-search-algorithm/) takes O(logn) time to perform the search in sorted arrays consisting of n elements.
* It takes O(logn) time to perform the search in [**Binary Search Tree**](https://www.gatevidyalay.com/binary-search-trees-data-structures/)consisting of n elements.

### **Drawback-**

 The main drawback of these techniques is-

* As the number of elements increases, time taken to perform the search also increases.
* This becomes problematic when total number of elements become too large.

## **Hashing in Data Structure-**

In data structures,

* Hashing is a well-known technique to search any particular element among several elements.
* It minimizes the number of comparisons while performing the search.

### **Advantage-**

 Unlike other searching techniques,

* Hashing is extremely efficient.
* The time taken by it to perform the search does not depend upon the total number of elements.
* It completes the search with constant time complexity O(1).

## **Hashing Mechanism-**

 In hashing,

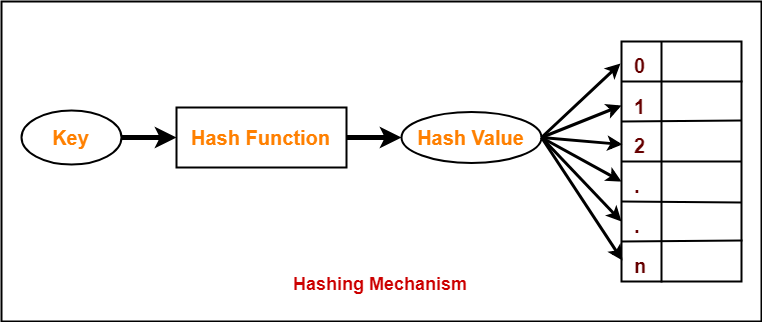
* An array data structure called as **Hash table** is used to store the data items.
* Based on the hash key value, data items are inserted into the hash table.

**Hash table**

Hash table is a type of data structure which is used for storing and accessing data very quickly. Insertion of data in a table is based on a key value. Hence every entry in the hash table is defined with some key. By using this key data can be searched in the hash table by few key comparisons and then searching time is dependent upon the size of the hash table.

**Hash Key Value**

* Hash key value is a special value that serves as an index for a data item.
* It indicates where the data item should be be stored in the hash table.
* Hash key value is generated using a hash function.



## **Hash Function**

|  |
| --- |
| Hash function is a function that maps any big number or string to a small integer value. |

* Hash function takes the data item as an input and returns a small integer value as an output.
* The small integer value is called as a hash value.
* Hash value of the data item is then used as an index for storing it into the hash table.

## **Types of Hash Functions**

 There are various types of hash functions available such as-

1. Division Hash Function
2. Mid Square Hash Function
3. Folding Hash Function etc

 It depends on the user which hash function he wants to use.

**Division method**

In this the hash function is dependent upon the remainder of a division. For example:-if the record 52,68,99,84 is to be placed in a hash table and let us take the table size is 10.

**Then:**

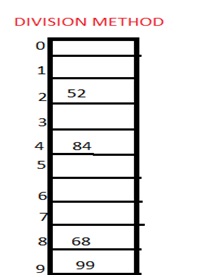
h(key)=record% table size.

2=52%10

8=68%10

9=99%10

4=84%10



**2. Mid square method**

In this method firstly key is squared and then mid part of the result is taken as the index. For example: consider that if we want to place a record of 3101 and the size of table is 1000. So 3101\*3101=9616201 i.e. **h (3101) = 162 (middle 3 digit)**

**3. Digit folding method**

In this method the key is divided into separate parts and by using some simple operations these parts are combined to produce a hash key. For example: consider a record of 12465512 then it will be divided into parts i.e. 124, 655, 12. After dividing the parts combine these parts by adding it.

H(key)=124+655+12

=791

## **Properties of Hash Function**

The properties of a good hash function are-

* It is efficiently computable.
* It minimizes the number of collisions.
* It distributes the keys uniformly over the table.
* The hash function should generate different hash values for the similar string.
* The hash function is easy to understand and simple to compute.

To gain better understanding about Hashing in Data Structures,

**RELEVANT READING MATERIAL AND REFERENCES:**

**Source Notes:**

1. <https://www.gatevidyalay.com/hashing/>
2. <https://www.includehelp.com/data-structure-tutorial/hashing.aspx>

**Lecture Video:**

1. <https://www.youtube.com/watch?v=SWG9uoYUvSc>

**Online Notes:**

1. <http://www.crectirupati.com/sites/default/files/lecture_notes/ds%20ln.pdf>
2. <http://www.vssut.ac.in/lecture_notes/lecture1428550942.pdf>

**Text Book Reading:**

1. Cormen, Leiserson, Rivest, Stein, “*Introduction to Algorithms*”, Prentice Hall of India, 3rd edition 2012. problem, Graph coloring.
2. Lipschutz, S., “*Data Structures, Schaum's Outline Series*”, Tata McGraw Hill.

**Online Book Reference:**

1. <https://www.edutechlearners.com/download/books/DS.pdf>

**In addition: PPT can be also be given.**