Hashing

Team : Aimers

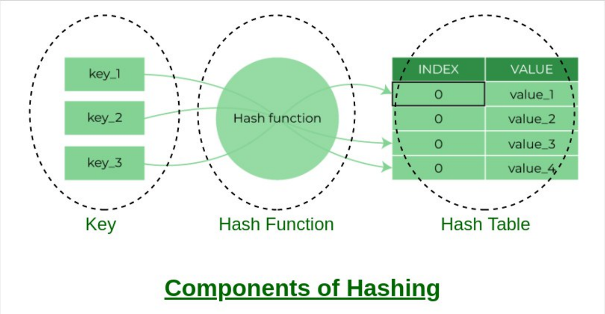
Mathematics and Scientific Computing
MNNIT ALLAHABAD

April 21, 2023

Hashing (1/3)

- Hashing is a technique used to store and retrieve data in a database very quickly and efficiently.
    - Hashing involves mapping data to a unique value, called a hash code.
    - The hash code is then used to index into an array, where the data is stored.
    - To retrieve the data, the hash code is simply re-computed and used to index into the array.

Hashing (2/3)



file:Hashing.jpeg

- It is fundamental to many data structures, such as hash tables and hash trees.
- This allows for quick look-up when searching for a specific value, as well as easy identification of any duplicates.

Hashing (3/3)

- Hashing is an efficient way to store and retrieve data in a data structure as it avoids the need for comparisons between elements.

Hash Table

- **Hash table** is a data structure in which keys are mapped to array positions by a hash function.
- In a hash table, an element with key k is stored at index h(k) not k. It means a hash function, h is used to calculate the index at which the element with key k will be stored. This process of mapping the keys to appropriate locations (or indices) in a hash table is called **hashing**.
- **Collision** is a situation when two or more keys map to the same memory location.

Hash Function

- A **hash function** is a mathematical formula which, when applied to a key, produces an integer which can be used as an index for the key in the hash table.
- The main aim of a hash function is that elements should be relatively, randomly, and uniformly distributed.
- It produces a unique set of integers within some suitable range in order to reduce the number of collisions.
- In practice, there is no hash function that eliminates collisions completely. A good hash function can only minimize the number of collisions by spreading the elements uniformly throughout the array.

Different Hash Functions

In this section, we will discuss some of the popular hash functions which help to minimize collision.

- Division Method
- Mid-Square Method
- Multiplication Method
- Folding Method

Division Method

It is the most simple method of hashing an integer x. This method divides x by M and then uses the remainder obtained. In this case, the hash function can be given as

$$h(x) = x \mod M$$

Example

Lets calculate the hash values of keys 1234 and 5462.
Setting M = 97, hash values can be calculated as:
h(1234) = 1234
h(5642) = 5642

Multiplication Method (1/3)

The steps involved in the multiplication method are as follows:

1. Choose a constant A such that $0 < A < 1$.
2. Multiply the key k by A.
3. Extract the fractional part of kA.
4. Multiply the result of Step 3 by the size of hash table (m).

Multiplication Method (2/3)

Hence, the hash function can be given as:

$$h(k) = \lfloor m(kA \mod 1) \rfloor$$

Example

Multiplication Method (3/3)

Given a hash table of size 1000, map the key 12345 to an appropriate location in the hash table. We will use $A = 0.618033, m = 1000,$and$k = 12345$

$$h(12345) = \lfloor 1000(123450.618033 mod 1) \rfloor$$
$$h(12345) = \lfloor 1000(7629.617385 mod 1) \rfloor$$
$$h(12345) = \lfloor 1000(0.617385) \rfloor$$
$$h(12345) = \lfloor 617.385 \rfloor$$
$$h(12345) = 617$$

Mid-square method (1/2)

The mid-square method is a good hash function which works in two steps:

1. Square the value of the key. That is, find k^2.
2. Extract the middle r digits of the result obtained in Step 1

Hence, the hash function can be given as:

$$h(k) = s$$

where s is obtained by selecting r digits from k^2.

Example

Mid-square method (2/2)

Lets calculate the hash value for keys 1234 and 5642. The hash table has 100 memory locations.

Note that the hash table has 100 memory locations whose indices vary from 0 to 99. This means that only two digits are needed to map the key to a location in the hash table, so

$r = 2.$

When $k = 1234, k^2 = 1522756, h(1234) = 27$

When $k = 5642, k2 = 31832164, h(5642) = 21$

Observe that the 3rd and 4th digits starting from the right are chosen

Folding method (1/2)

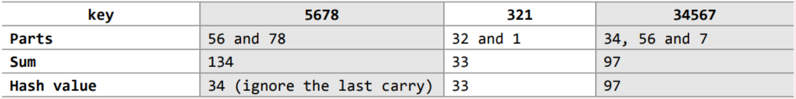
The folding method works in the following two steps:

1. Divide the key value into a number of parts. That is, divide k into parts $k_1, k_2, ..., k_n$, where each part has the same number of digits except the last part which may have lesser digits than the other parts.
2. Add the individual parts. That is, obtain the sum of $k_1 + k_2 + ... + k_n$
3. The hash value is produced by ignoring the last carry, if any.

Folding method (2/2)

Example

Given a hash table of 100 locations, calculate the hash value using folding method for keys 5678, 321, and 34567.
Since there are 100 memory locations to address, we will break the key into parts where each part (except the last) will contain two digits. The hash values can be obtained as shown below:



key	5678	321	34567
Parts	56 and 78	32 and 1	34, 56 and 7
Sum	134	33	97
Hash value	34 (ignore the last carry)	33	97

Collision

As discussed earlier, collision occurs when the hash function maps two different keys to the same location. Obviously, two records cannot be stored in the same location. Therefore,
a method used to solve the problem of collision, also called **collision resolution technique**, is applied.
The two most popular methods of resolving collisions are:

1. Open addressing [1]
    1. Linear Probing
    2. Quadratic Probing
    3. Double Hashing
2. Chaining

Uses

Some real-life examples [2] of hashing in data structure include -

- In Libraries - A library contains an endless supply of books. The librarian assigns each book a unique number. This distinctive number aids in locating the exact position of the books on the bookshelf.
- In Schools - Each student in a class is assigned a unique roll number for easy identification. The school authority, later on, uses the unique roll number to retrieve relevant information about the particular student.
- Security Purposes - One must take precautions to ensure that the account does not end up in the wrong hands when they first visit a website that requests authentication through "Sign Up" and where they submit the login information to access the personal accounts. As a result, the database stores the entered password as a hash.

References

- Reema Thareja
  Data structures using C.
  Oxford University Press, 2014.
- Hashing and Application.
  https://dhawal-khapre19.medium.com/