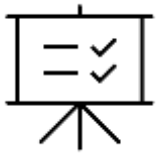
Advanced Algorithms & Data Structures

AIM OF THE SESSION

To familiarize students with the basic concept of Hashing

INSTRUCTIONAL OBJECTIVES



This Session is designed to:

1. Demonstrate Hashing. Separate chaining
2. Describe the types of hashing
3. Describe each method

LEARNING OUTCOMES

At the end of this session, you should be able to:

1. Understand the fundamental concepts of hashing, including hash functions, hash tables
2. Learns different hashing techniques (e.g., Division Method, Mid Square Method, Digit Folding Method, Multiplication Method) and their suitability for different scenarios.
3. Apply hashing to solve practical problems involving efficient data retrieval and storage.

Introduction to Hashing

- Hashing is a technique or process of mapping keys, and values into the hash table by using a hash function.
- It is done for faster access to elements and minimizes the number of comparisons while performing the search.
- The efficiency of mapping depends on the efficiency of the hash function used.

Hashing Mechanism

- It's a technique for efficiently storing and retrieving data using a hash function.
- A hash function converts a key (like a name or a number) into an index, which is used to locate the corresponding value in a data structure called a hash table.
- Ideally, each key maps to a unique index, but collisions (multiple keys mapping to the same index) can occur and need to be resolved.

Introduction to Hashing

Hash Table

- It's the primary data structure used for hashing.
- It's typically an array of slots, where each slot can hold a key-value pair.
- The hash function determines which slot a key-value pair is placed into.

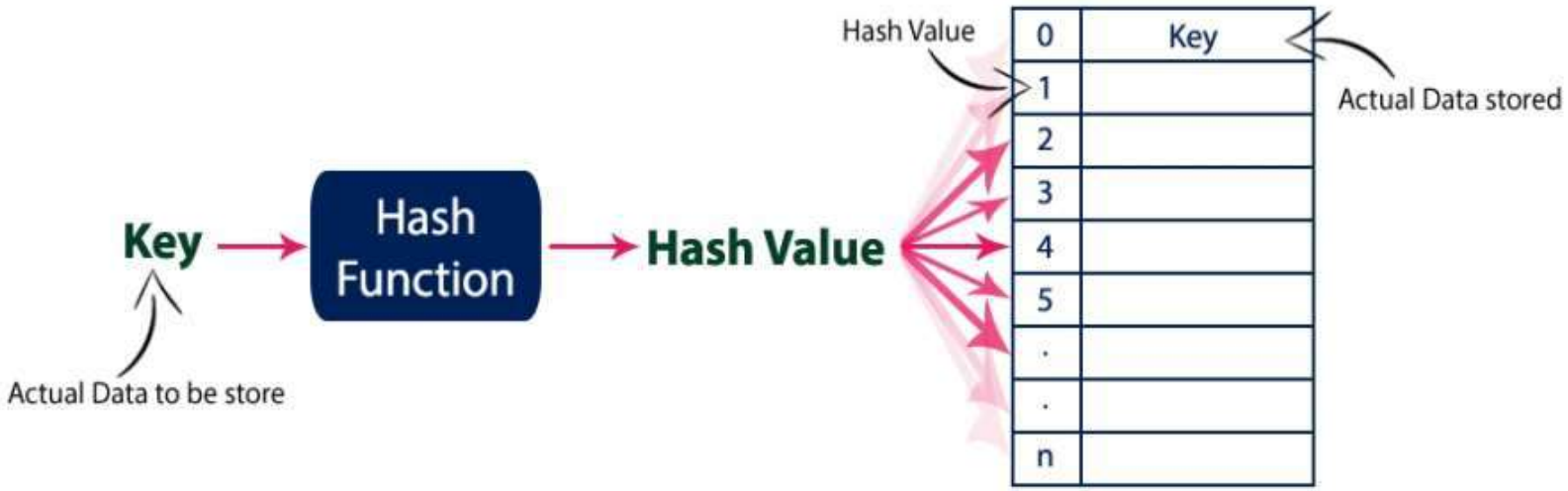
Hash Function

- It's the core component of hashing.
- It takes a key as input and produces a unique integer value (the hash value) that serves as the index in the hash table.
- Common hash functions include division-based, multiplication-based, and universal hashing.

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 stored in the hash table.
- Hash key value is generated using a hash function.

Types of Hashing



Types of Hashing

1. Division Method
2. Mid Square Method
3. Digit Folding Method
4. Multiplicative Method

Division method

1. Division method

- It's a simple and common technique for creating hash functions.
- It involves dividing the key by a fixed integer value (usually the size of the hash table) and using the remainder as the hash value.
- The hash function for the Division method is:

  h(key) = Key% Table size.

Example: If the record 52,68,99,84 is to be placed in a hash table and the table size is 10.

- The hash function for the Division method is:

  h(key) = Key% table size.

  h(52) = 52%10 = 2

  h(68) = 68%10 = 8

  h(99) = 99%10 = 9

  h(84) = 84%10 = 4

0	
1	
2	52
3	
4	84
5	
6	
7	
8	68
9	99

Mid square method

2. **Mid square method**

- It's a technique for generating hash values.
- It involves squaring the key value and extracting a fixed number of digits from the middle of the squared result as the hash value.
- The hash function for the Mid Square method is:

  h(key) = Key*Key

Example: Consider a Key = 12

The hash function for the Mid Square method is:

h(key) = Key*Key

H(12) = 12 * 12 = 144

Extract middle digit (hash table size = 10)

Hash value = 4

Store or retrieve data at index 4 in the hash table

Digit folding method

3. Digit folding method

- It's a hash function method that involves folding (dividing) a key into multiple parts, adding those parts, and using the result as the hash value.

Example: Consider a Key = 123456789 with a hash table size of 1000

Now, break the key into segments of equal length i.e., 123 | 456 | 789.

Sum these digits within each segment.

H(key)=123+456+789

=1368

If the sum exceeds the hash table size, use modulo operation to fit it within the table's range.

In the example: 1368 % 1000 = 368 (the final hash value).

Multiplicative Method

4. Multiplicative Method

- Multiplicative hashing is a technique for mapping keys to indices in a hash table that uses multiplication and the fractional part of a number to achieve a good distribution of keys.
- The steps involved in the multiplication method are as follows:
  - Choose a constant A ($0 < A < 1$)
    - A common choice is the golden ratio ($A \approx 0.6180339887$)
  - Multiply the key k by A
  - Extract the fractional part of kA
  - Multiply the fractional part by the size of the hash table M
  - Take the floor of the result

Multiplicative Method

- The hash function for the Multiplicative method is:

  h(k) =floor [m(kA mod size)]

Example: Consider a Key k: 23

Donald knuth suggested to use Constant A: 0.6180339887

Hash table size m: 10

key is 23

h(23)=floor(10*(23*0.618033))%10

=floor(142.14)%10

=142%10

=2

So key 23 is stored at location 2 of hash table.

IMPORTANT FACTS RELATED TO THE SESSION

Advantages of Hashing

- Hashing is extremely efficient.
- It completes the search with constant time complexity O(1).

Disadvantages of Hashing

- Hash tables have a limited capacity and can fill up over time.
- Hash functions may not have the ability to navigate to the next or previous data set.

Applications/Uses of Hashing in real life

- Password storage: Hashing is used to store passwords securely by applying a hash function to the password and storing the hashed result, rather than the plain text password.
- Data compression: Hashing is used in data compression algorithms, such as the Huffman coding algorithm, to encode data efficiently.

EXAMPLES

1. Let us consider a simple hash function as “key mod 10” and sequence of keys as 40, 70, 76, 85, 92, 73, 100. Draw hash table and use separate chaining when collision occurs.

2. Construct a hash table for the given input sequence: 461, 137, 675, 197, 294, 965, 131 and hash function H(k) = k mod 10. Use separate chaining if collision occurs.

SELF-ASSESSMENT QUESTIONS

What is the hash function used in the division method?

a) h(k) = k/m
b) h(k) = k mod m
c) h(k) = m/k
d) h(k) = m mod k

2. Using division method, in a given hash table of size 157, the key of value 172 be placed at position

a) 19
b) 72
c) 15
d) 17

SUMMARY

- Hashing is a powerful technique for fast data access and storage, but understanding its limitations and careful implementation are essential for optimal performance.
- It minimizes the number of comparisons while performing the search.
- It depends on the user, which hash function we wants to use.

TERMINAL QUESTIONS

1. Define Hashing?
2. List out different types of Hashing Techniques?
3. Write a C program to implement hashing using separate chaining. Assume that the has table has a size of 10
4. Construct hash table using separate chaining for the following elements of table size 15, design your own hash function. 23,27,98,6,17,51,77,72,36,53.

REFERENCES FOR FURTHER LEARNING OF THE SESSION

Reference Books

1. Mark Allen Weiss, Data Structures and Algorithm Analysis in C, 2010 , Second Edition, PearsonEducation.
2. 2. Ellis Horowitz, Fundamentals of Data Structures in C: Second Edition, 2015
3. A.V.Aho, J. E. Hopcroft, and J. D. Ullman, “Data Structures And Algorithms”, Pearson Education, First Edition Reprint2003.

Sites and Web links:

1. https://nptel.ac.in/courses/106102064
2. https://in.udacity.com/course/intro-to-algorithms--cs215
3. https://www.coursera.org/learn/data-structures?action=enroll

THANK YOU