

Hashing

Hashing is a technique used in data structures that efficiently stores and retrieves data in a way that allows for quick access.

- Hashing involves mapping data to a specific index in a hash table (an array of items) using a hash function. It enables fast retrieval of information based on its key.
- The great thing about hashing is, we can achieve all three operations (search, insert and delete) **in $O(1)$ time on average.**

Components of Hashing

There are majorly three components of hashing:

1. Key
2. Hash Function
3. Hash Table

What is Collision?

The situation where the newly inserted key maps to an already occupied, and it must be handled using some collision handling technology.

Collision Resolution Techniques-

(1)Open Addressing:

- Linear Probing: Check the next available slot sequentially.
- Quadratic Probing: Use a quadratic function to find the next slot.
- Double Hashing: Use a second hash function for probing.

(2)Chaining:

- Each index in the table stores a linked list of keys that hash to the same index.

Example of hashing-

A hash function $h(k) = k \bmod 10$ is used to store a set of keys $\{12, 25, 35, 45, 26\}$ in a hash table of size 10. The insertion follows **linear probing** to resolve collisions.

Step 1- Insert 12

- $h(12)=12$
- $12 \bmod 10=2$
- Place 12 at index 2.

Hash table—[_, _, 12, _, _, _, _, _, _]

Step 2- Insert 25

- $h(25)=25$
- $25 \bmod 10=5$
- Place 25 at index 5.

Hash table—[_, _, 12, _, _, 25, _, _, _, _]

Step 3- Insert 35

- $h(35)=35$
- $35 \bmod 10=5$
- Place 35 at index 5
- **Collision** (Index 5 is already occupied by 25)
- Use **linear probing** → Check next available index **6**
- Place **35** at index **6**.

Hash table—[_, _, 12, _, _, 25, 35, _, _, _]

Step 4- Insert 45

- $h(45)=45$
- $45 \bmod 10=5$
- Place 35 at index 5
- **Collision** (Index 5 is occupied by 25)
- Check next available index **6** → **Occupied by 35**
- Check next available index **7** → **Empty**
- Place **45** at index **7**.

Hash table—[_, _, 12, _, _, 25, 35, 45, _, _]

Step 5- Insert 26

- $h(26)=26$
- $26 \bmod 10=6$
- **Collision-** (Index 6 is occupied by 35)
- Check next available index 7 → **Occupied by 45**
- Check next available index 8 → **Empty**
- Place **26** at index **8**.

Final Hash table—[_, _, 12, _, _, 25, 35, 45, 26, _]

