



ECAP770

ADVANCE DATA STRUCTURES

Ashwani Kumar
Assistant Professor

Learning Outcomes



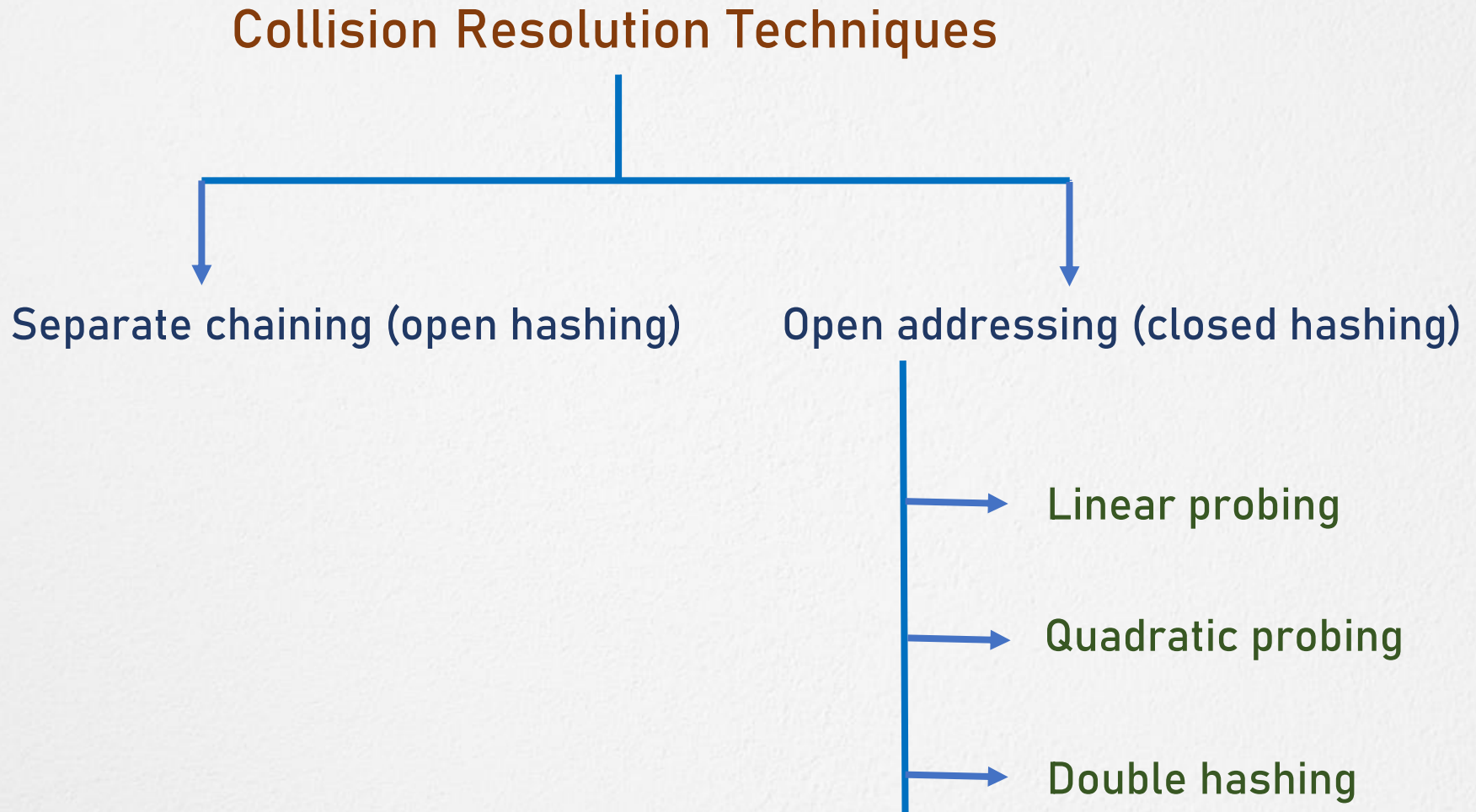
After this lecture, you will be able to

- understand Collision Resolution Techniques

Collision Resolution

- When two keys or hash values compete with a single hash table slot, then Collision occur.
- To resolve collision we use collision resolution techniques.
- Collisions can be reduced with a selection of a good hash function.

Collision Resolution Techniques



Separate Chaining

- In this technique, a linked list is created from the slot in which collision has occurred, after which the new key is inserted into the linked list.
- This linked list of slots looks like a chain, so it is called separate chaining.

Performance of Chaining

Load factor $\alpha = n/m$

m = Number of slots in hash table

n = Number of keys to be inserted in hash table

Expected time to search = $O(1 + \alpha)$

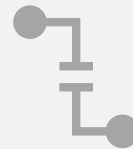
Expected time to delete = $O(1 + \alpha)$

Time to insert = $O(1)$

Time Complexity

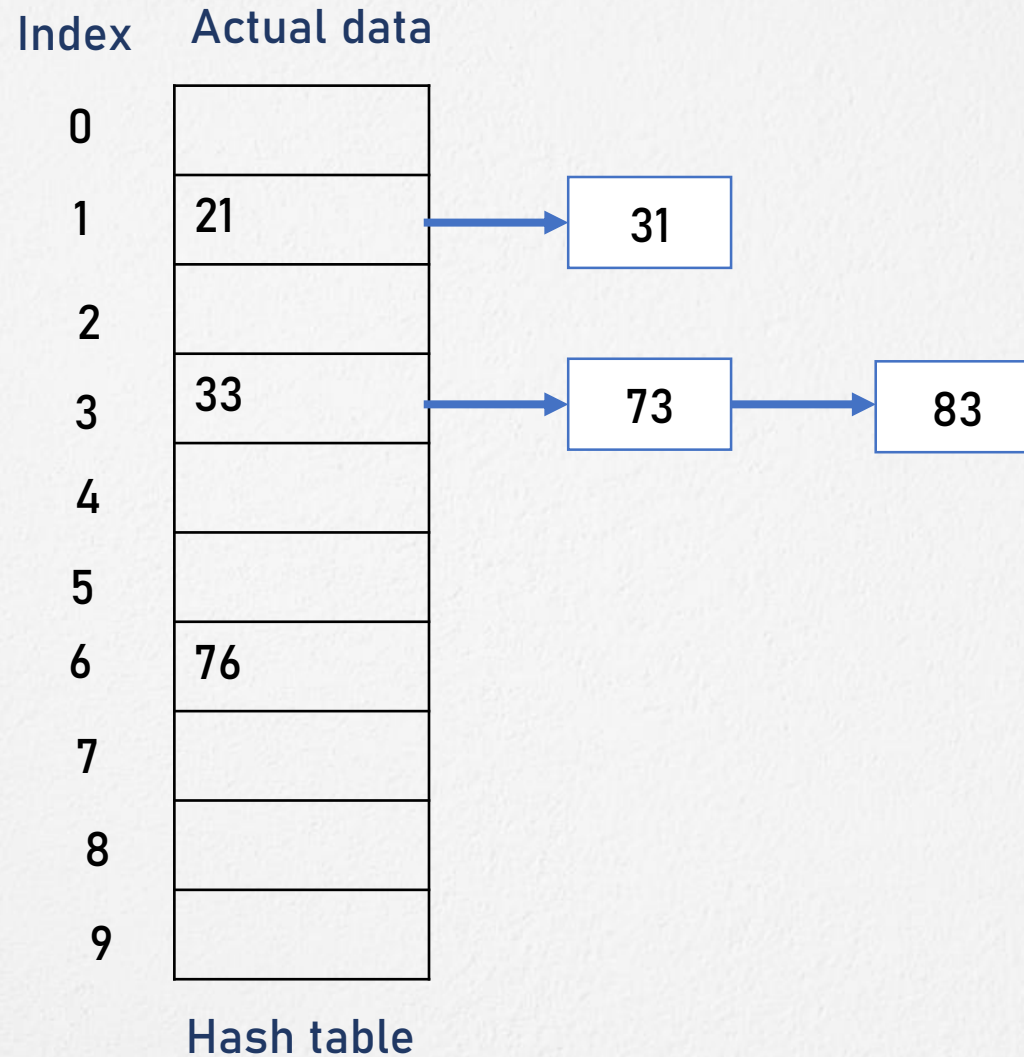


Worst case complexity for searching is $O(n)$.

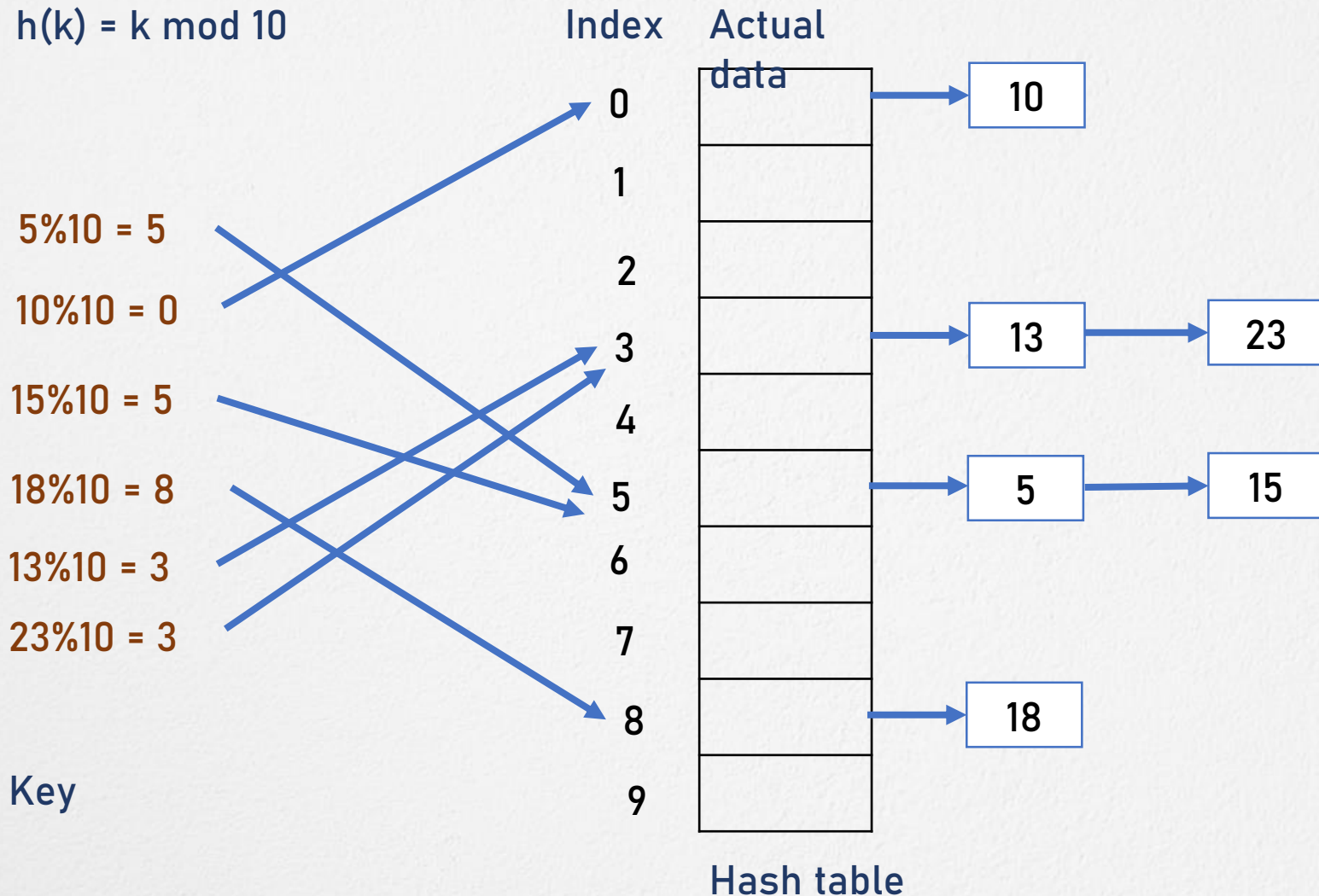


Worst case complexity for deletion is $O(n)$.

Separate chaining



Example: Separate chaining



Advantages of Separate Chaining

- It is easy to implement.
- The hash table never fills full, so we can add more elements to the chain.
- It is mostly used when it is unknown how many and how frequently keys may be inserted or deleted.
- It is less sensitive to the function of the hashing.

Disadvantages of Separate Chaining

- The cache performance of chaining is not good.
- The memory wastage is too much in this method.
- It requires more space for element links.
- If the chain becomes long, then search time can become $O(n)$ in the worst case.



That's all for now...