## KEY POINTS

Hash Maps:

Used to store key value pairs which can be accessed in O(1) time.

Achieved by hashing a value to compute the address e.g.

addr = key mod size

One can also use the folding method to calculate the address.

Collisions can be avoided by using open or closed addr.

load factor = items stored / size of HT

When the l.f. reaches a threshold we can resize the HT.

open ⇒ find next slot

closed ⇒ linked list for each key.

## NAME/DATE/SUBJECT

# Hash Tables

## NOTES

### ADT

- Stores key-value pairs in an array. Look ups take place in mostly constant time.
- This happens by calculating the index to store the KVP in, allowing for random searches.

memory location = key value mod n

Hashing algorithm: a function f: key → addr.

The folding method is another way to divide the key into equal parts and then add the parts together.

e.g. phone # 0145 2834 5654

⇒ 01 + 45 + 28 + 34 + 56 + 54 = 218

You may divide by some constant and take the remainder, depending on the size of the table.

## SUMMARY

KEY POINTS

NAME/DATE/SUBJECT

# Collisions - Open Addressing

NOTES

Collisions occur when two keys take the same index (from the hashing function).

Open addressing: placing an item elsewhere than it should be.

Linear probing - placing it in the next available spot.

This will use a linear search to keep searching for the next spot. If there is no space by the time you reach the end, then you wrap to the beginning of the list.

↳ This applies for locating clashed objects! You start at the index and keep searching.

Another way to reduce the number of clashes is to allocate more memory than required.

$$\text{Load factor} = \frac{\text{number of items stored}}{\text{size of array}}$$

e.g. LF = 70%

You could use a threshold to decide when the hash table should be resized.

L.P may lead to clustering - all the information gets stuck in one part of the map.

SUMMARY

KEY POINTS

NAME/DATE/SUBJECT

# Open Addressing

NOTES

Plus 3 Rehash: We look at every third place along when looking for free slots.

Quadratic probing: look at (failed attempts)$^2$ along each time,

Double hashing: the result of the second hash will describe the number of positions to shift when looking.

## HASH FUNCTION GOALS:

- All elements have a unique position.
- The elements use all the space available.

↑ These are perfect conditions

Realistic goals:

- Minimise collisions
- Uniform distributions
- The HF will be easy to calculate, and provides good collis. resolution.

SUMMARY

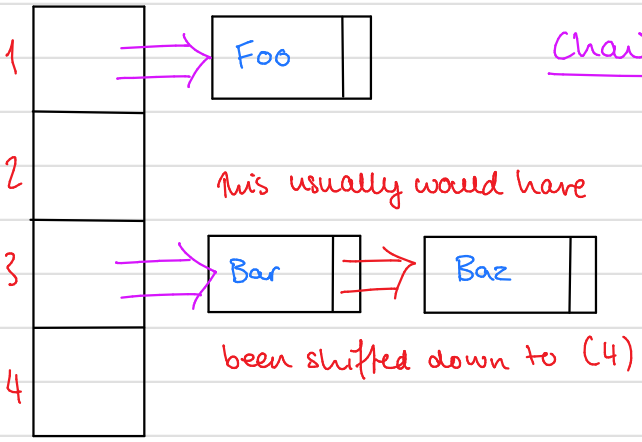
KEY POINTS

NAME/DATE/SUBJECT

Collisions - closed addressing

NOTES

Closed Addressing: when each key in the map is a pointer to the head of a linked list.




Chaining: You will need to calculate the index, then traverse the linked list in order to locate the value.

Chaining is typically better than worst case array traversal; however, if the load factor is small then linear probing may be faster.

SUMMARY