

1.4 DATA TYPES & STRUCTURES · 1.4.2

Hash tables

Hashing, collisions, load factor & fast lookup. Spec 1.4.2(b)(c).

01 The idea

What Stores key–value pairs; hash function gives the **address**.

Hash fn $\text{address} = \text{key} \text{ MOD } \text{tableSize}$

Lookup Hash the key → go **straight** to the cell (fast).

Example $23 \text{ MOD } 10 = 3 \rightarrow$ stored at address 3.

02 Collisions

What Two keys hash to the **same address**.

Probing Put item in the **next free cell**; search successive cells.

Chaining Each cell holds a **list**; add colliding items to it.

Example $33 \text{ MOD } 10 = 3$ (taken) → cell 4.

03 Search, delete & load factor

Search Hash key → check cell → successive cells until found or a **blank**.

Delete Use a **"deleted" marker** so searches don't stop early.

Load factor $\text{items} \div \text{table size} =$ how full it is.

Effect High load factor → more collisions → **slower**. Resize/rehash.

FINAL PASS BEFORE THE EXAM

Rapid exam tips

Six slips on hash-table questions.

01

Hash the key to **calculate** the address — don't scan the table.

02

Two keys, same address = **collision**. Always show how it's resolved.

03

Probing → **next free cell**; chaining → **list** per cell.

04

Search stops at a **blank** cell = "not present".

05

Deleting needs a **marker**, or later searches miss items.

06

Load factor = **items ÷ size**; high = more collisions = slower.