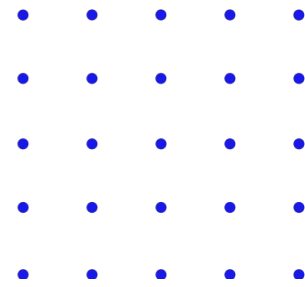


Introduction to Data Structures



By: Yehia M. Abu Eita

Outlines

- Introduction
- Why data structures are important?
- Types of data structures
- Static and dynamic data structures
- Space and time complexities
- Some data structures applications

Introduction

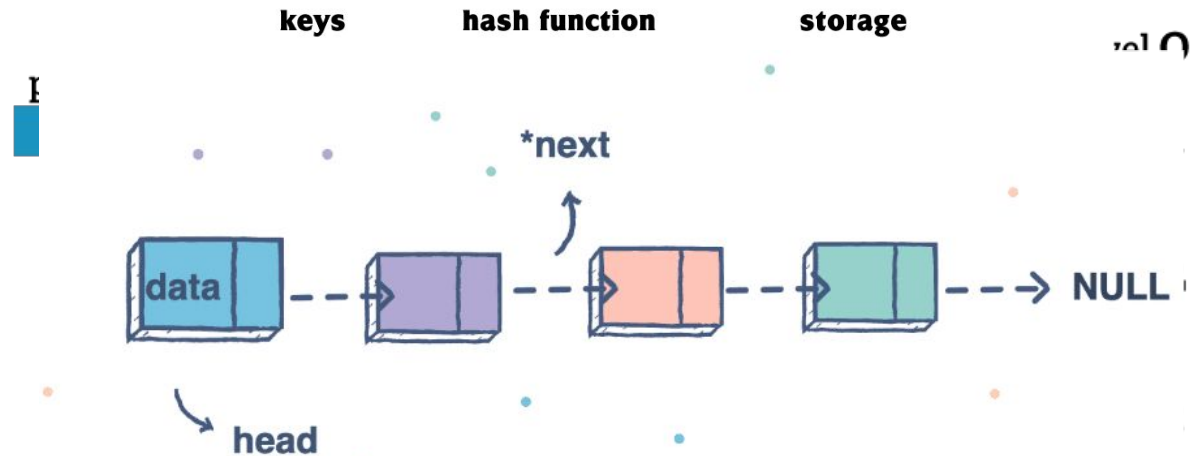
- It is data **organization, management, and storage formats** that enables **efficient access** and **modifications** of data.
- It allows the processing of a **large amount** of data in a relatively **short** period of **time**.
- The main purpose of using data structures is to **reduce time** and **space complexities**.
- An **efficient** data structure makes use of **minimum memory** space and takes the **minimal** possible **time** to modify data.

Why data structures are important?

- It is the building block of more **complex problems**.
- It **enhances software design** and algorithms' implementation.
- Choosing an inappropriate data structure may result in slow runtimes or inefficient code.
- Data structures are used in:
 - **Storing data**
 - **Managing resources and services**
 - **Data exchange**

Types of data structures

- Arrays
- Stack
- Heap
- Queue
- Tree
- Table
- Graph
- Linked lists



Static and dynamic data structures

- **Static** data structure:
 - These types of data structures have **fixed size** in memory.
 - Data can be modified but the allocated memory **can not be changed** during the **run-time**.
 - **Arrays** are static data structures and any data structure is implemented using arrays is static also.
- **Dynamic** data structure:
 - These types of data structures have **dynamic size** in memory.
 - Both data and the allocated memory **can be modified** during the **run-time**.
 - **Linked lists** are dynamic data structures and any data structure is implemented using linked lists is dynamic also.

Space and time complexities

- **Space complexity:**

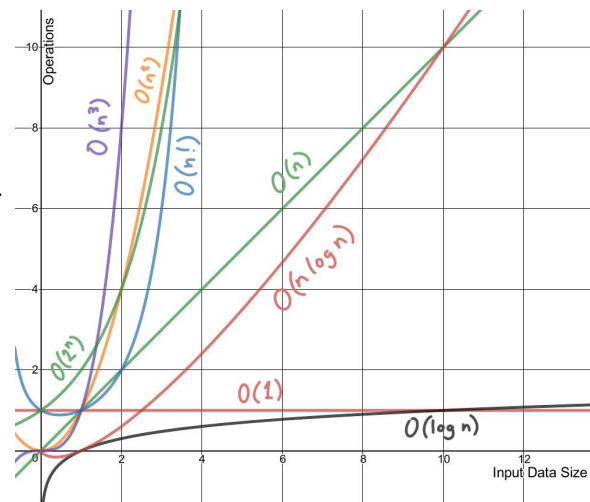
- It means **how much space of memory** is consumed during the **run-time**.
- Space complexity is **increased** as long as you define **more variables** that allocates more memory.

- **Time complexity:**

- It means **how much time** does it take **to finish** the needed operation.
- Time complexity is **increased** as long as you make **more iterations/steps** to reach your goal.

Space and time complexities

- Complexity is measured in Big-O notation.
- Time complexity is CPU operations Vs. data size.
 - **$O(1)$:**
 - This means there is **no dependency** on the input data size (**The best**).
 - **$O(\log n)$:**
 - This means there is a **logarithmic increase** in operations.
 - **$O(n)$:**
 - This means that number of operations is the **same** as the data size.
 - **$O(n \log n)$:**
 - This means that there is **more increase** in operations.
 - **$O(n^2)$:**
 - This means that number of operations is **increasing rapidly** with any small change in data size (**Very bad**).



Space and time complexities

Data structure	Time Complexity (Worst)				Space complexity
	Access	Search	Insertion	Deletion	Worst
Array	$O(1)$	$O(N)$	$O(N)$	$O(N)$	$O(N)$
Stack	$O(N)$	$O(N)$	$O(1)$	$O(1)$	$O(N)$
Queue	$O(N)$	$O(N)$	$O(1)$	$O(1)$	$O(N)$
Singly Linked list	$O(N)$	$O(N)$	$O(1)$	$O(1)$	$O(N)$
Doubly Linked List	$O(N)$	$O(N)$	$O(1)$	$O(1)$	$O(N)$

Some data structures applications

- Some **arrays** applications:
 - 2D arrays, commonly known as, matrices, are used in image processing.
 - It is also used in speech processing, in which each speech signal is an array.
 - Your viewing screen is also a multidimensional array of pixels.
- Some **linked** lists applications:
 - Used for symbol table management in a designing compiler.
 - Used in switching between applications and programs (Alt + Tab) in the Operating system (implemented using Circular Linked List).
 - It can be used to implement Stacks, Queues, Graphs, and Trees.

Some data structures applications

- Some **Stack** applications:
 - Undo/Redo button/operation in word processors.
 - Syntaxes in languages are parsed using stacks.
 - Message logs and all messages you get are arranged in a stack.
- Some **Queue** applications:
 - Operating System uses queues for job scheduling.
 - To handle congestion in the networking queue can be used.
 - Sending an email, it will be queued.

Summary

- Now you have good understanding about data structures.
- It's clear that data structures are involved in many applications.
- Choosing suitable data structure results in fast and memory saving applications.
- Time and space complexities are very important for data structures.