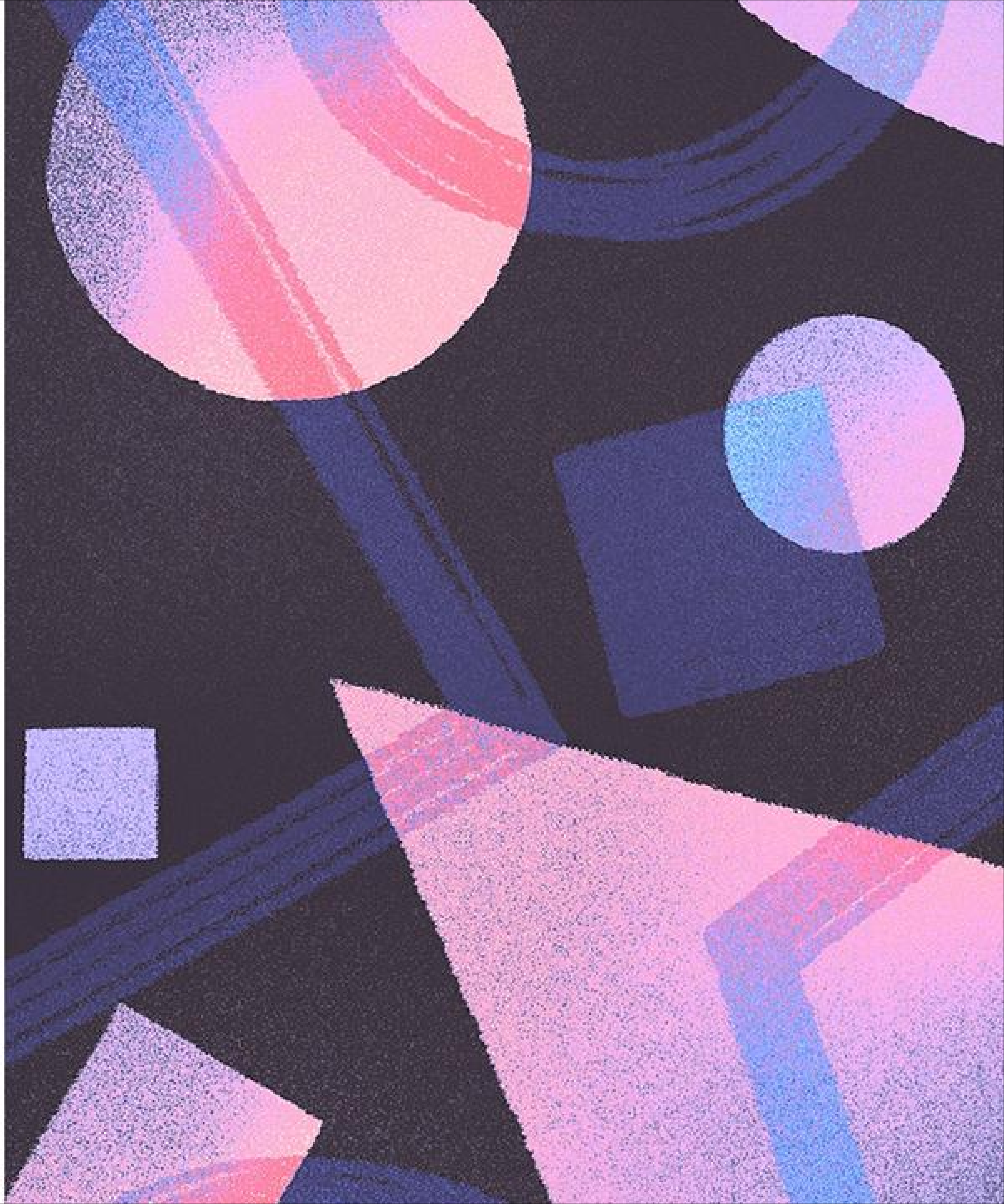
Data Structures Lab
Lecture 5

Stack & Queue

Maryam Skaik, Dec-2025



CONTENT

01

PART

Stack

02

PART

Queue

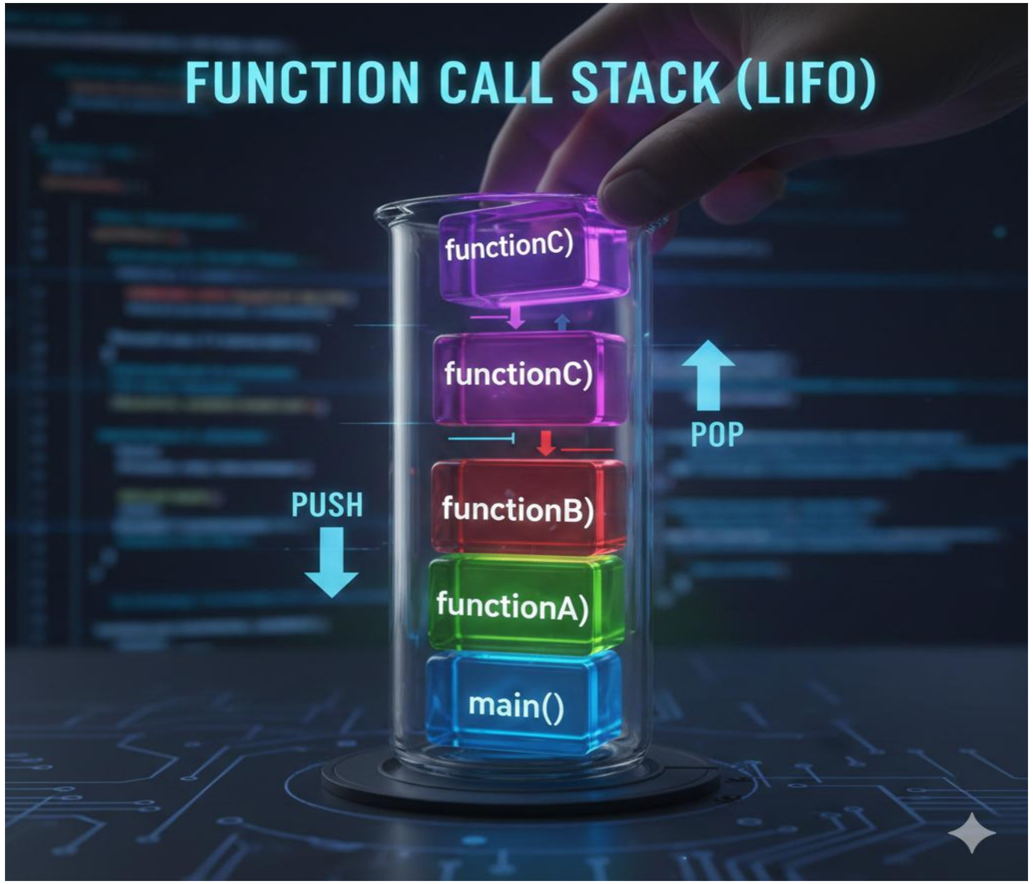
03

PART

Implementation

01: What Is a Stack?

- Stack Follows **LIFO** principle
  - Last In, First Out
- Operations happen at one end only (top)
- Real-life examples:
  - Stack of plates
  - Function call stack



Stack Operations

- **push(x)** → insert element
- **pop()** → remove top element
- **peek()** → read top element
- **isEmpty()** → check if stack is empty

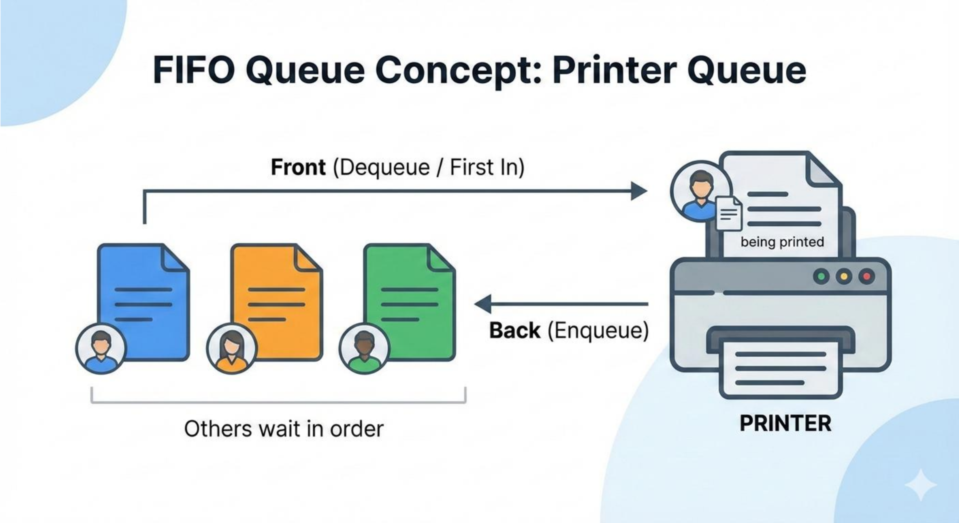
Important: No random access

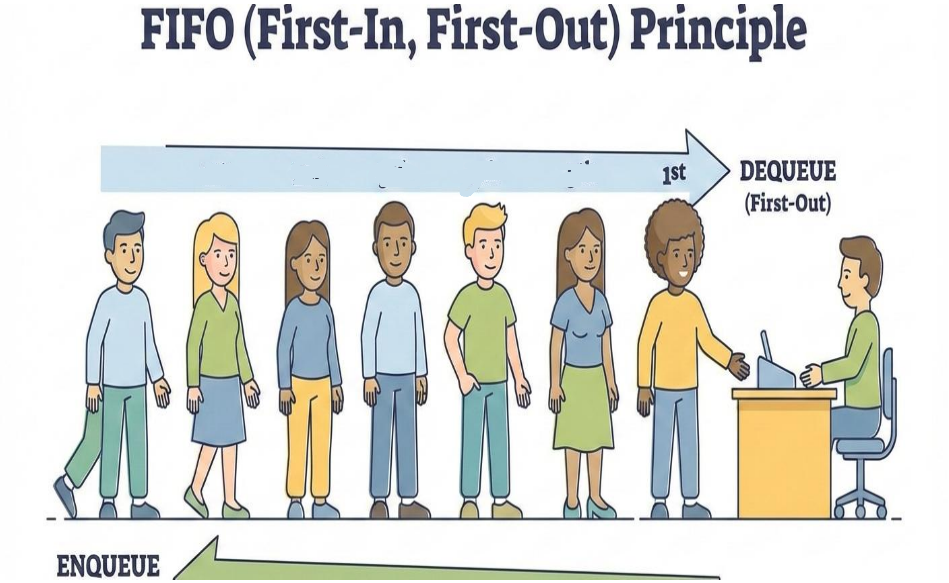
Stack as an ADT

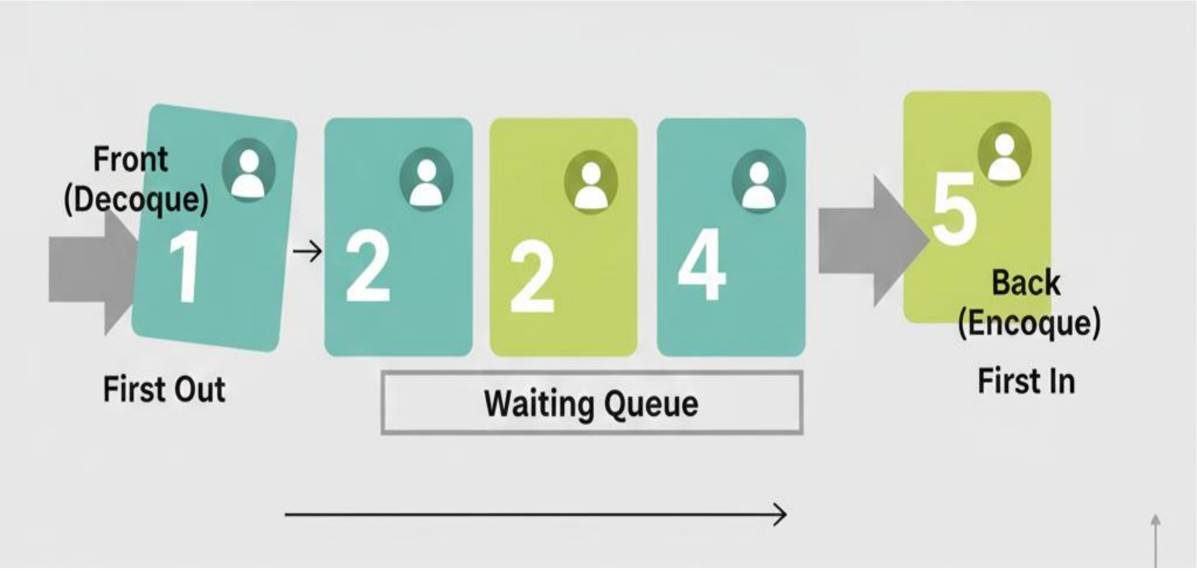
- **ADT = Abstract Data Type**
- Defines *what* operations do, not *how* they are implemented
- We use stack operations **without caring about implementation**

02: What Is a Queue?

- Follows FIFO principle
  - First In, First Out
- Insert at rear, remove from front
- Real-life examples:
  - Waiting line
  - Printer queue







Queue Operations

- **enqueue(x)** → insert element
- **dequeue()** → remove element
- **peek()** → front element
- **isEmpty()**

Key difference from stack:

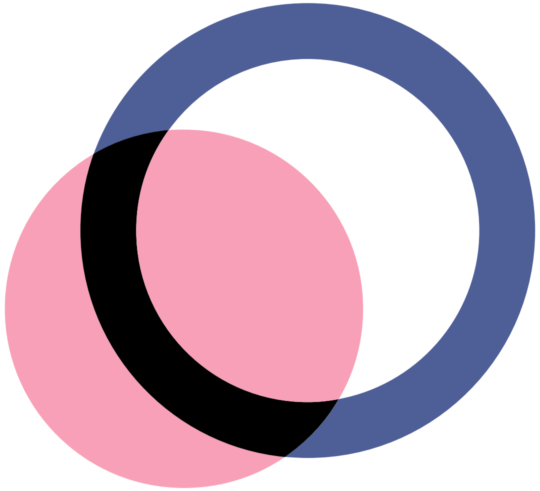
- Two ends are used

Queue as an ADT

- Defines behavior, not implementation
- Same queue operations
- Implementation can change

Stack vs Queue

Stack	Queue
LIFO	FIFO
One end	Two ends
Used in recursion	Used in scheduling



THANK YOU