មហាវិទ្យាល័យវិស្វកម្ម
FACULTY OF ENGINEERING

Data Structure & Algorithm

Lecture 7

Abstract Data Types: Queues and Priority Queues

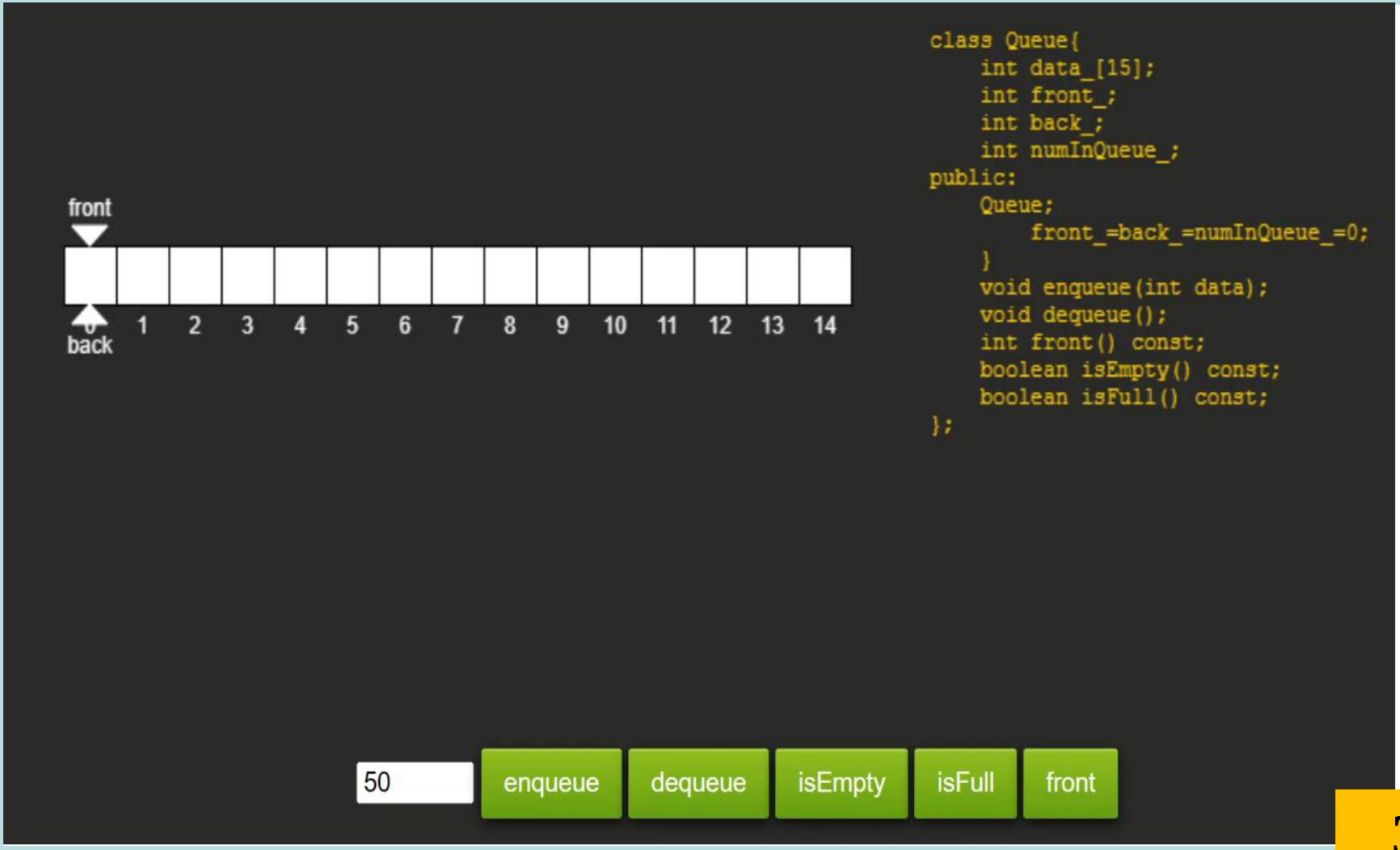
Chhoeum Vantha, Ph.D.

Telecom & Electronic Engineering

Content

- Abstract Data Types
  - Stacks
  - Queues and Priority Queues
  - Linked Lists
  - Abstract Data Types
  - Specialized Lists

What is the video all about?

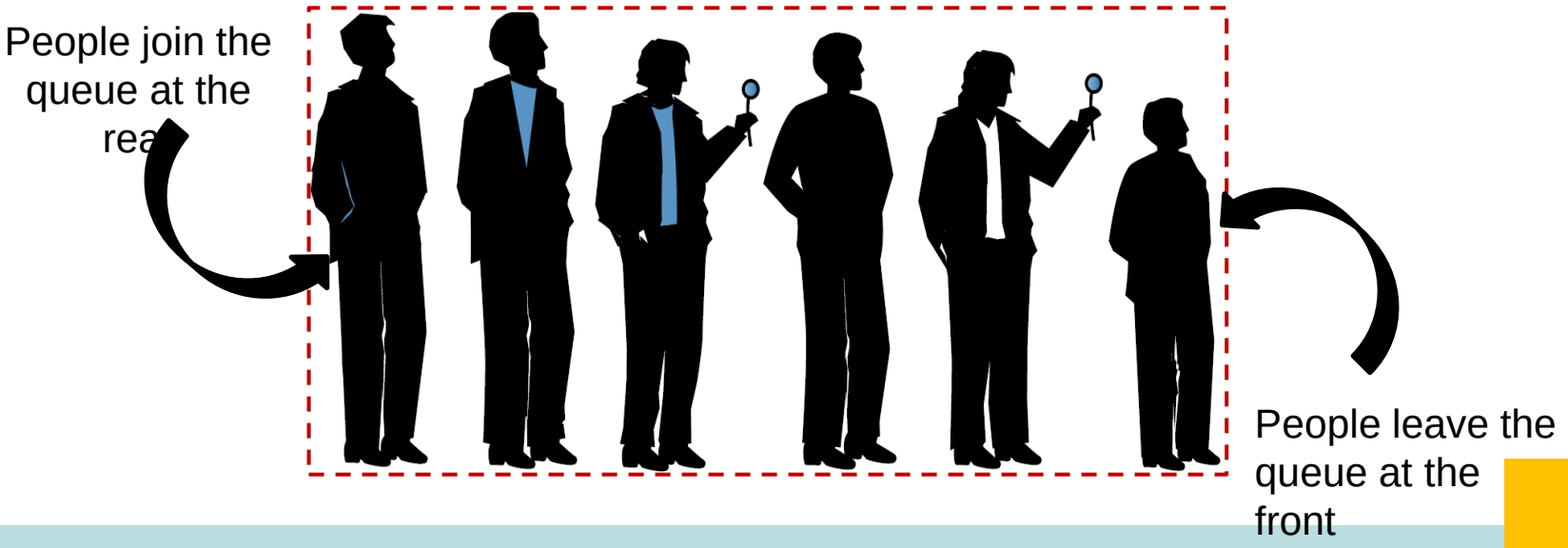


Queues

- The word queue is British for line (the kind you wait in)
- In computer science a queue is a data structure that is similar to a stack but in a queue the First In, the First Out (FIFO)

Queues

- A queue works like a line at the movies
- The first person to join the rear of the line is the first person to reach the front of the line and buy a ticket
- The last person to line up is the last person to buy a ticket

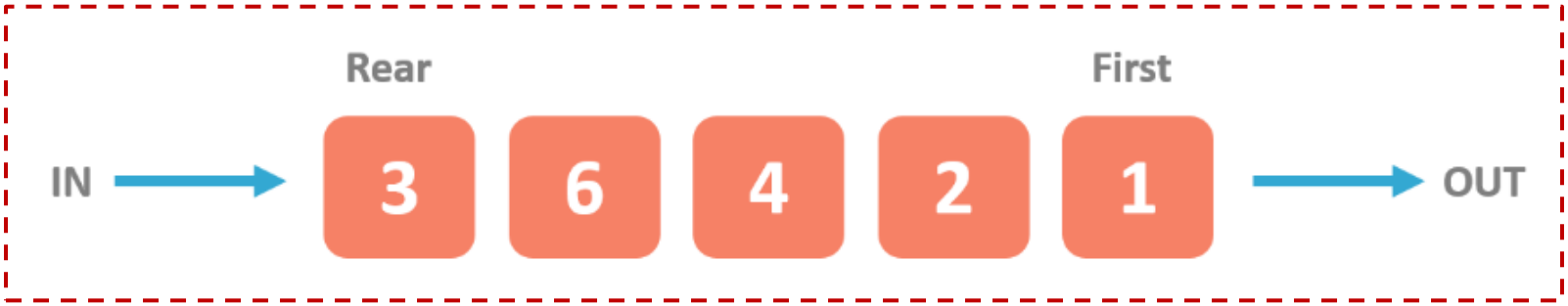


Types of Queues

- Simple Queue
- Circular Queue
- Priority Queue
- Double-Ended Queue (Deque)

Types of Queues: Simple Queue

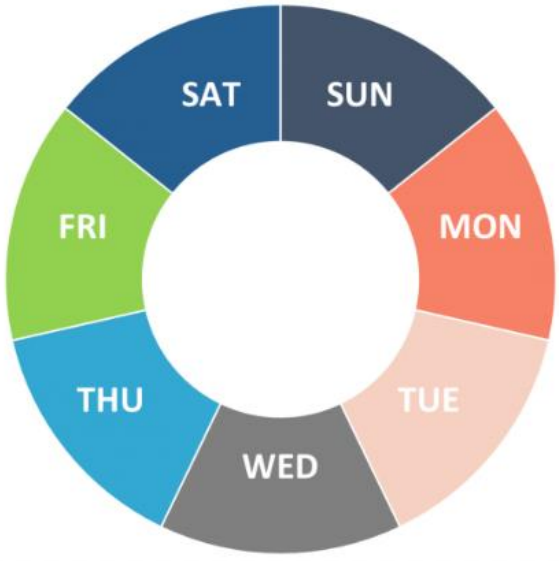
- It is the most basic queue in which the insertion of an item is done at the front of the queue and deletion takes place at the end of the queue.
- Ordered collection of comparable data kinds.
- Queue structure is FIFO (First in, First Out).



Types of Queues: Circular Queue

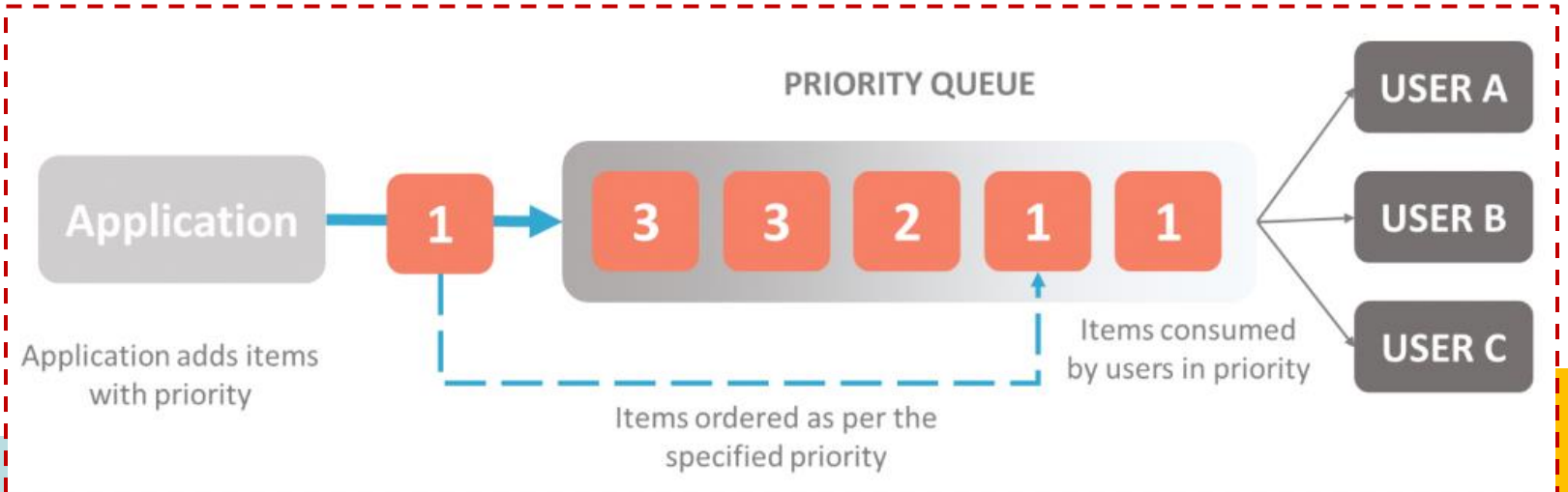
- A circular queue is a special case of a simple queue in which the last member is linked to the first.
- As a result, a circle-like structure is formed.
- The last node is connected to the first node.

- Insertion takes place at the front of the queue and deletion at the end of the queue.



Types of Queues: Priority Queue

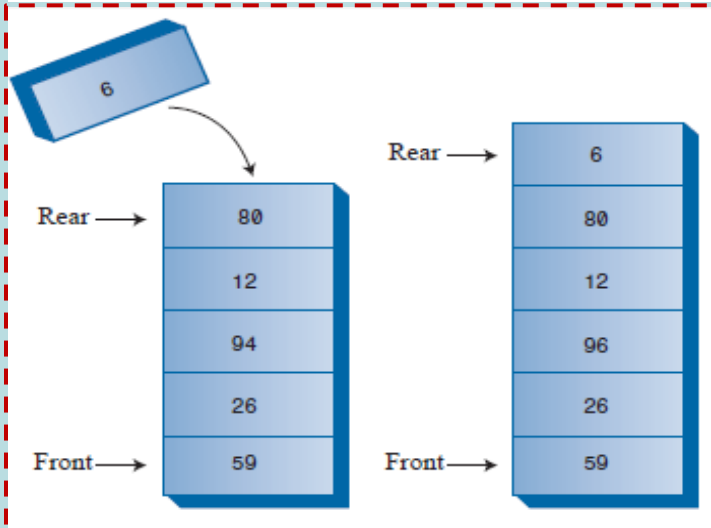
- In a priority queue, the nodes will have some predefined priority in the priority queue.
- The node with the least priority will be the first to be removed from the queue.
- Insertion takes place in the order of arrival of the nodes.



Types of Queues: Double-Ended Queue (Deque)

- In a double-ended queue, insertion and deletion can take place at both the front and rear ends of the queue.

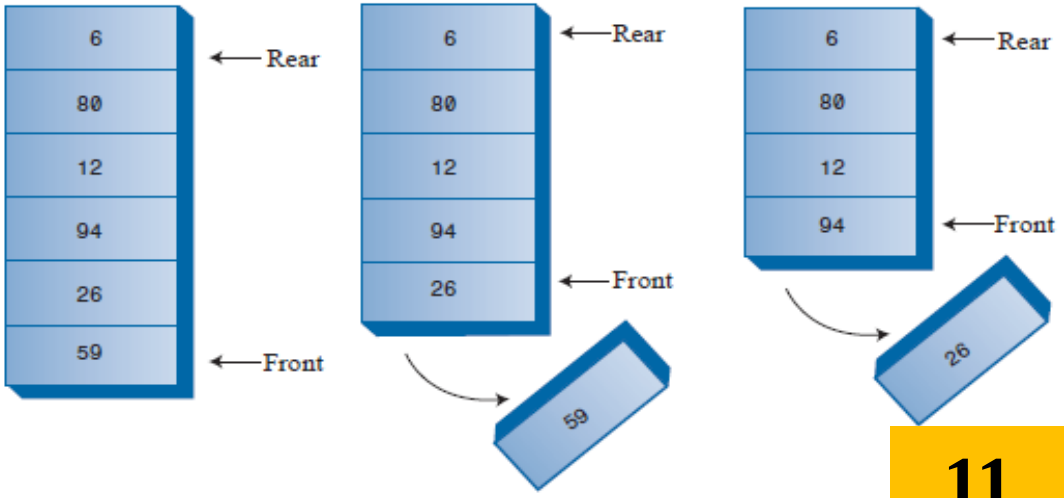
Queue: Insert and Remove Operations



New item inserted at rear of queue

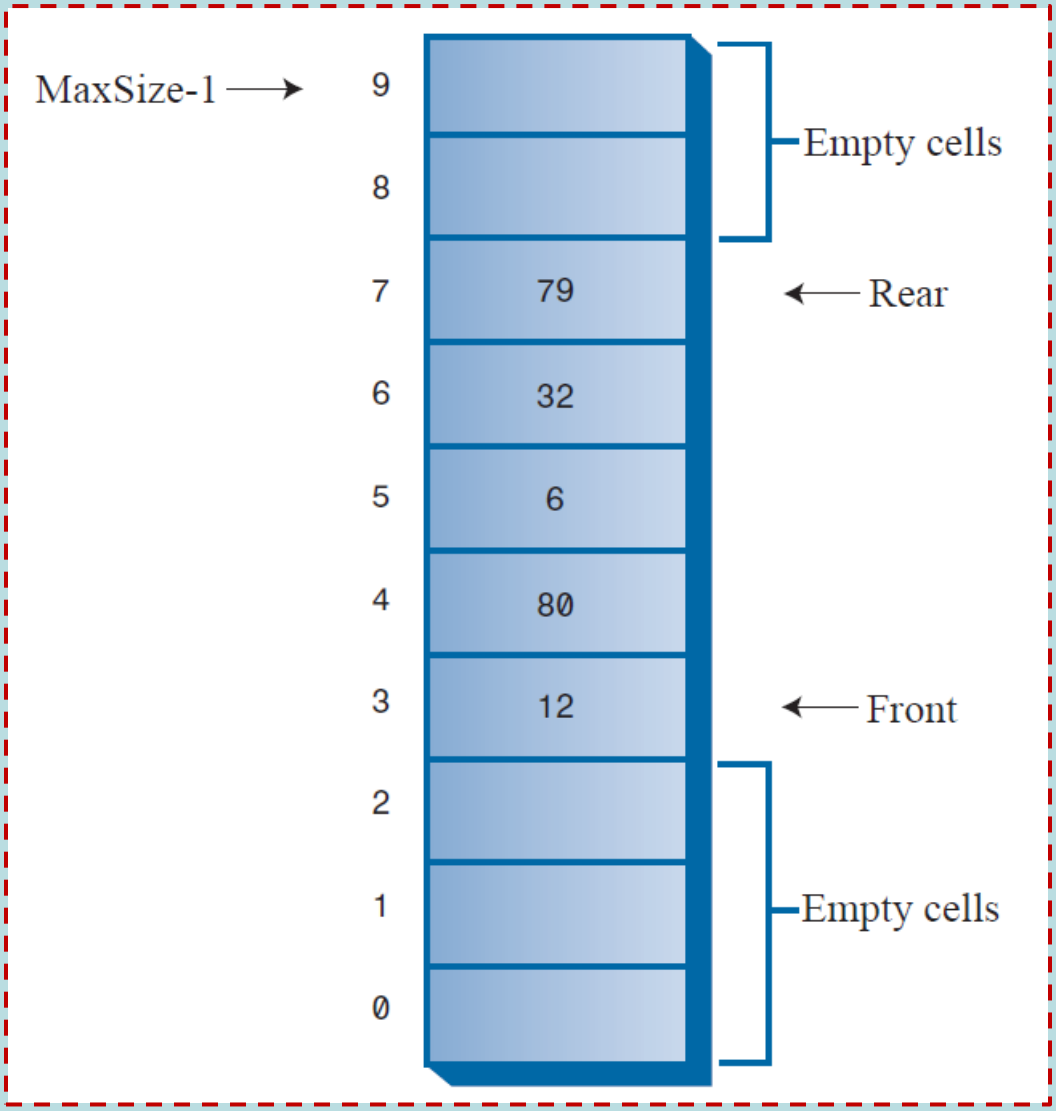
Add to Rear

Remove Front



Two items removed from front of queue

Queue: Some Items Removed



Queue: The Empty and Full Error

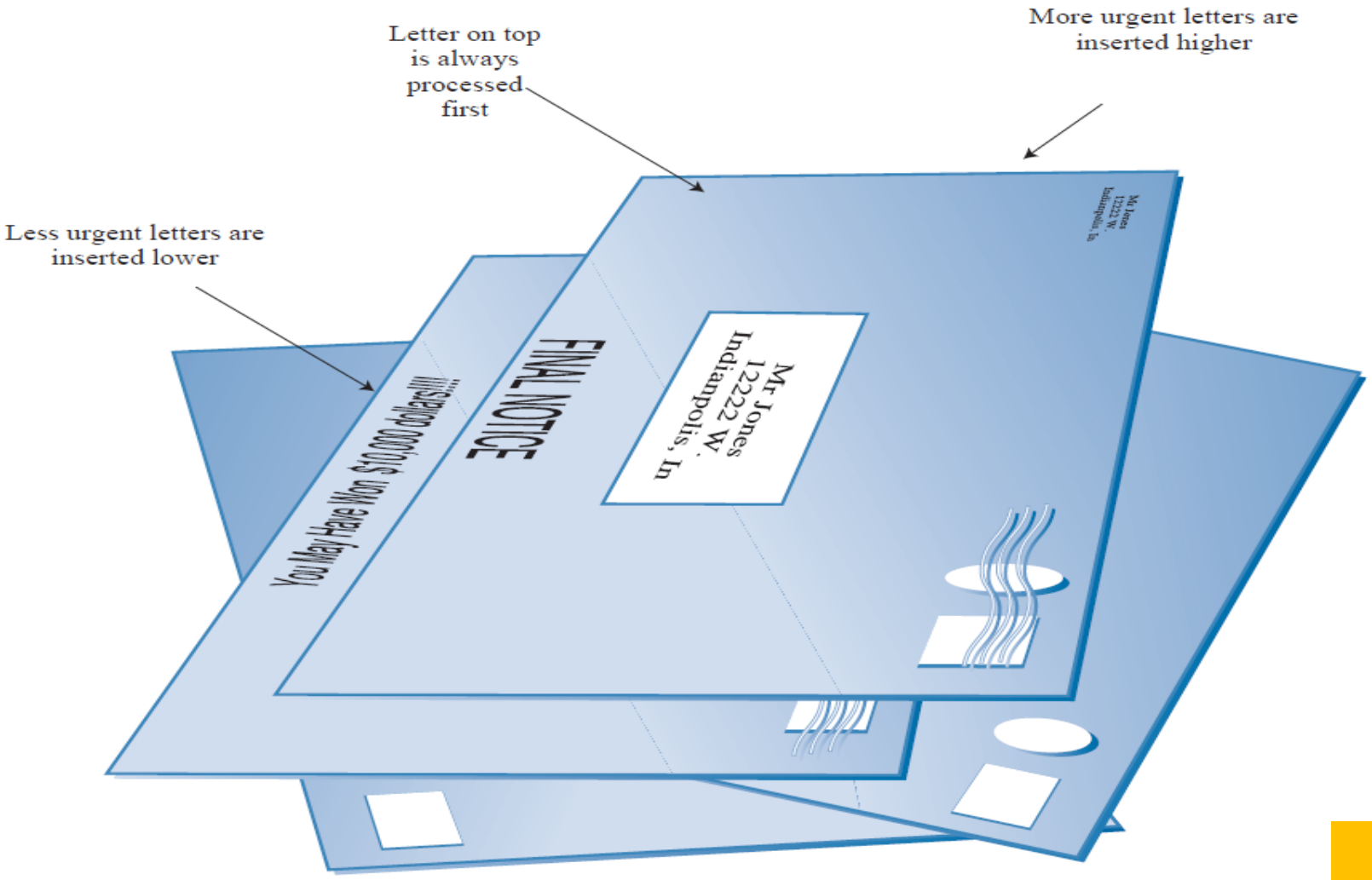
Case of Empty Error:

- In case, there are no more items in the queue, you cannot remove an item from queue

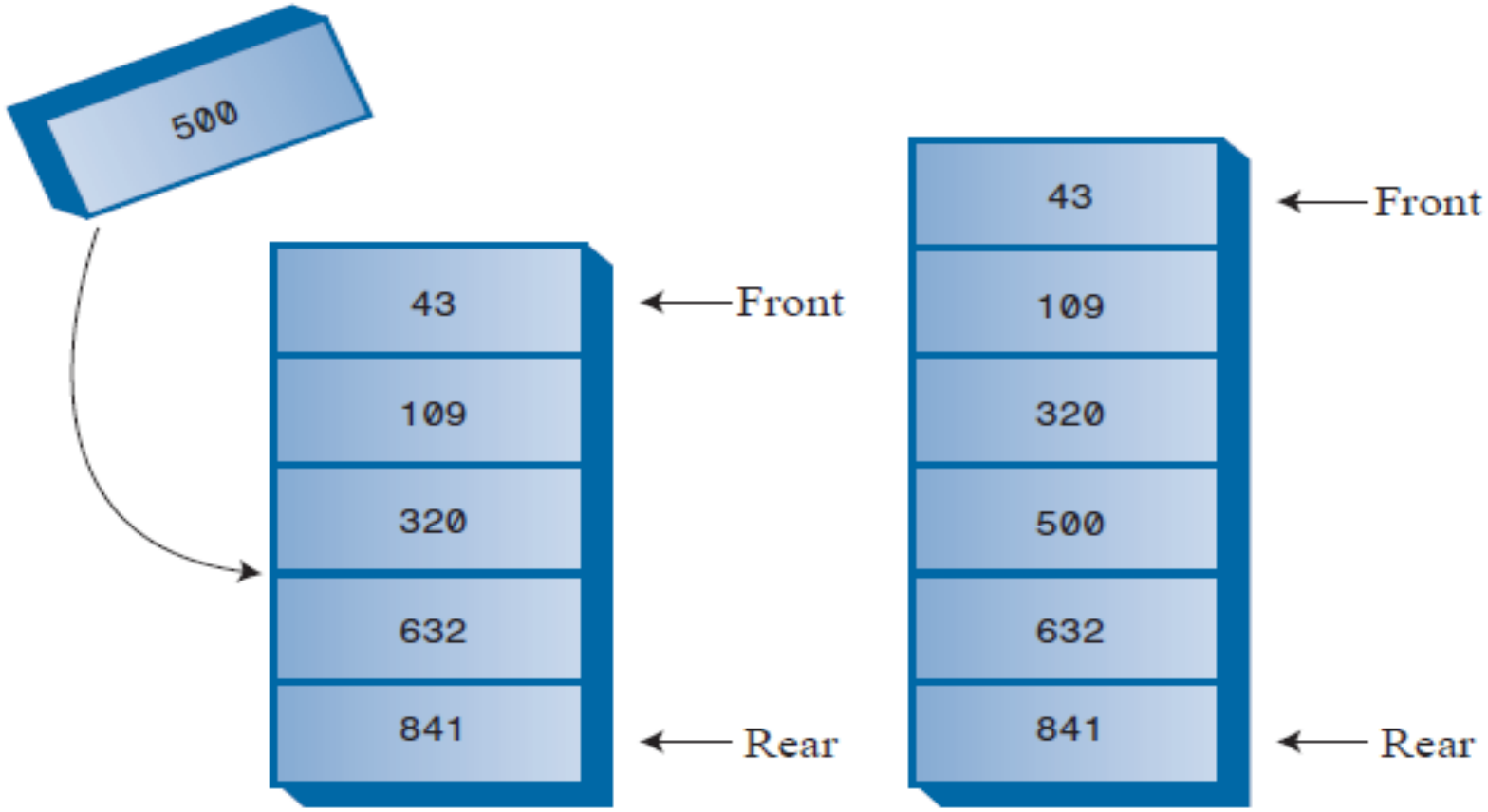
Case of Full Error:

- In case, all the cells are already occupied, you cannot insert a new item to queue

Priority Queues

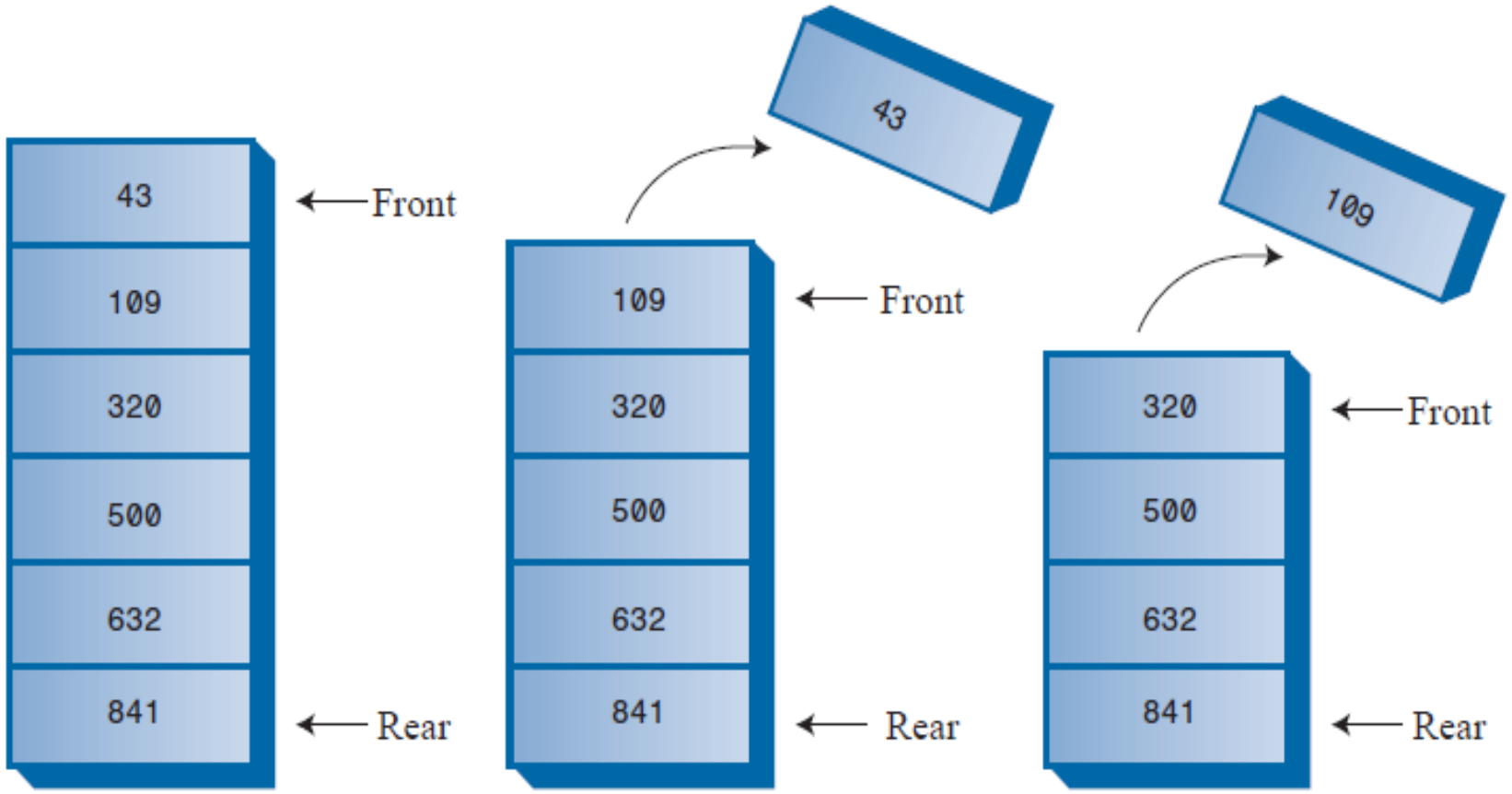


Priority Queues



New item inserted in priority queue

Priority Queues



Two items removed from front of priority queue

Priority Queues

- is a more specialized data structure than a stack or a queue
- It is useful tool in a surprising number of situations
- Like an ordinary queue, a priority queue has a front and a rear, and items are inserted in the rear and removed from the front

Priority Queues

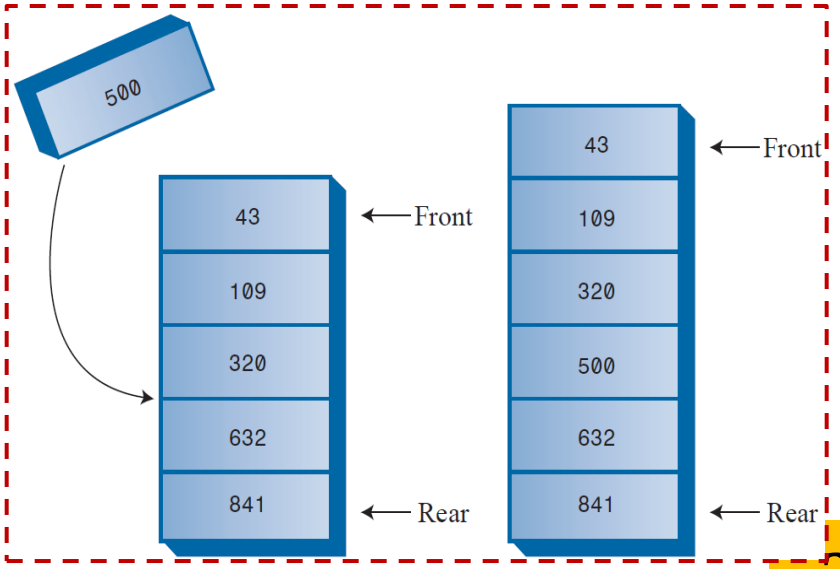
- A priority queue is a data structure that allows at least the following two operations:
- insert: inserts a data item into the priority queue.
- deleteMin: finds, returns, and removes the minimum element in the priority queue

Priority Queues

- In a priority queue, items are ordered by key value, so that the item with the lowest key (or highest key) is always at the front
- Items are inserted in the proper position to maintain the order

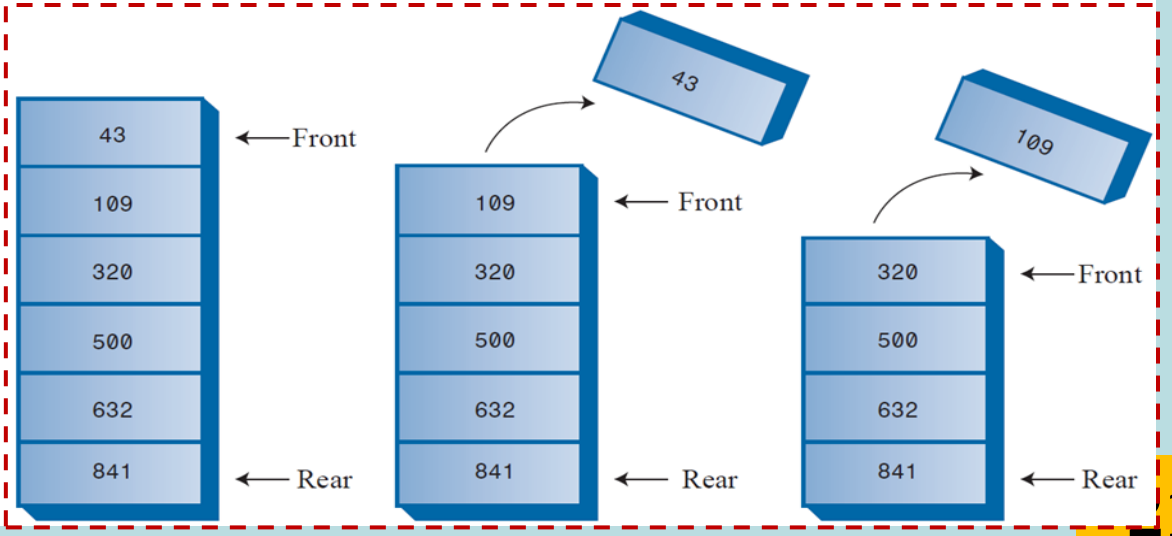
Priority Queues: Inserting a New Item

- First, find the appropriate position by his value, insert a new item to the found position
- Notice that there is no wraparound in implementation of the priority queue
- Insertion is slow because the proper in-order position must be found, but deletion is fast



Priority Queues: Deleting an Item

- The item to be removed is always the front item (in both ascending and descending), thus removal is quick and easy
- The item is removed and the Front moves to the next item of the array
- No comparisons or shifting are necessary



Queue vs Priority Queue

- Queue, the first-in-first-out rule is implemented
- Priority queue, the values are removed on the basis of priority.
- The element with the highest priority is removed first

applications of the queue in real-life are:

- People on an escalator
- Cashier line in a store
- A car wash line
- One way exits

W7 – Lab 7

Exercise 1

- Create a Queue using Standard Template Library (STL) with a few operations:
  - push(int newData)
  - pop()
  - size()
  - front()
  - back()

  And make a function (*showQ)* to show all elements in the queue

Exercise 2

- Create a class of Queue with full operations:
  - IsFull(),
  - InsertQ(int NewItem),
  - RemoveQ(),
  - IsEmptyQ(),
  - SizeQ(),
  - PeekFrontQ().

Exercise 3

- Create a class of Priority Queue (ascending and descending) with full operations:
  - IsFull(),
  - InsertPQ(int NewItem),
  - RemovePQ(),
  - IsEmptyPQ(),
  - SizePQ(),
  - PeekFrontPQ().

Midterm Choice:

1. Oral: 5 min
2. Explain code, Code-Tree steps
3. Code-Tree steps, Code-Flow chart
4. Flow Chart-Code
5. Theory: close book exam

Thanks!