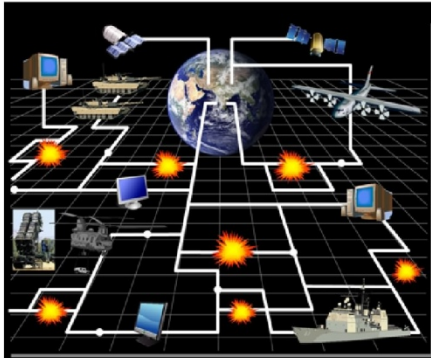


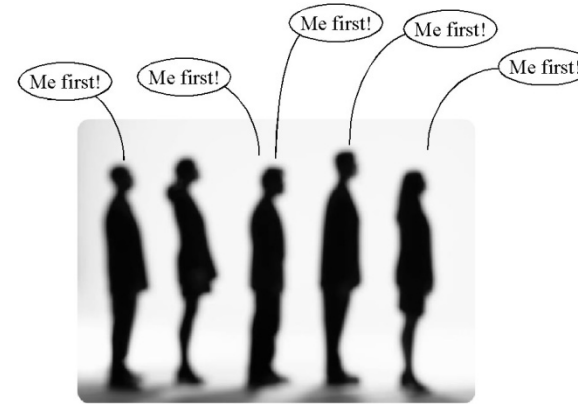
Queues

What are they? Where are they used?

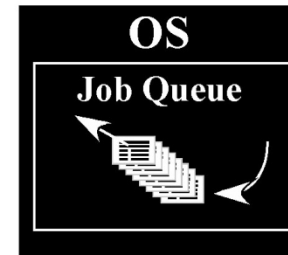
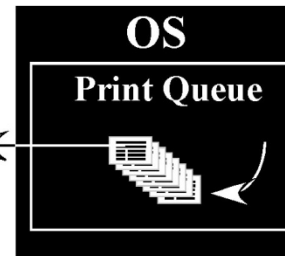
Queue - An abstract data type in which elements are added to the end and removed from the front.



Modeling and Simulation



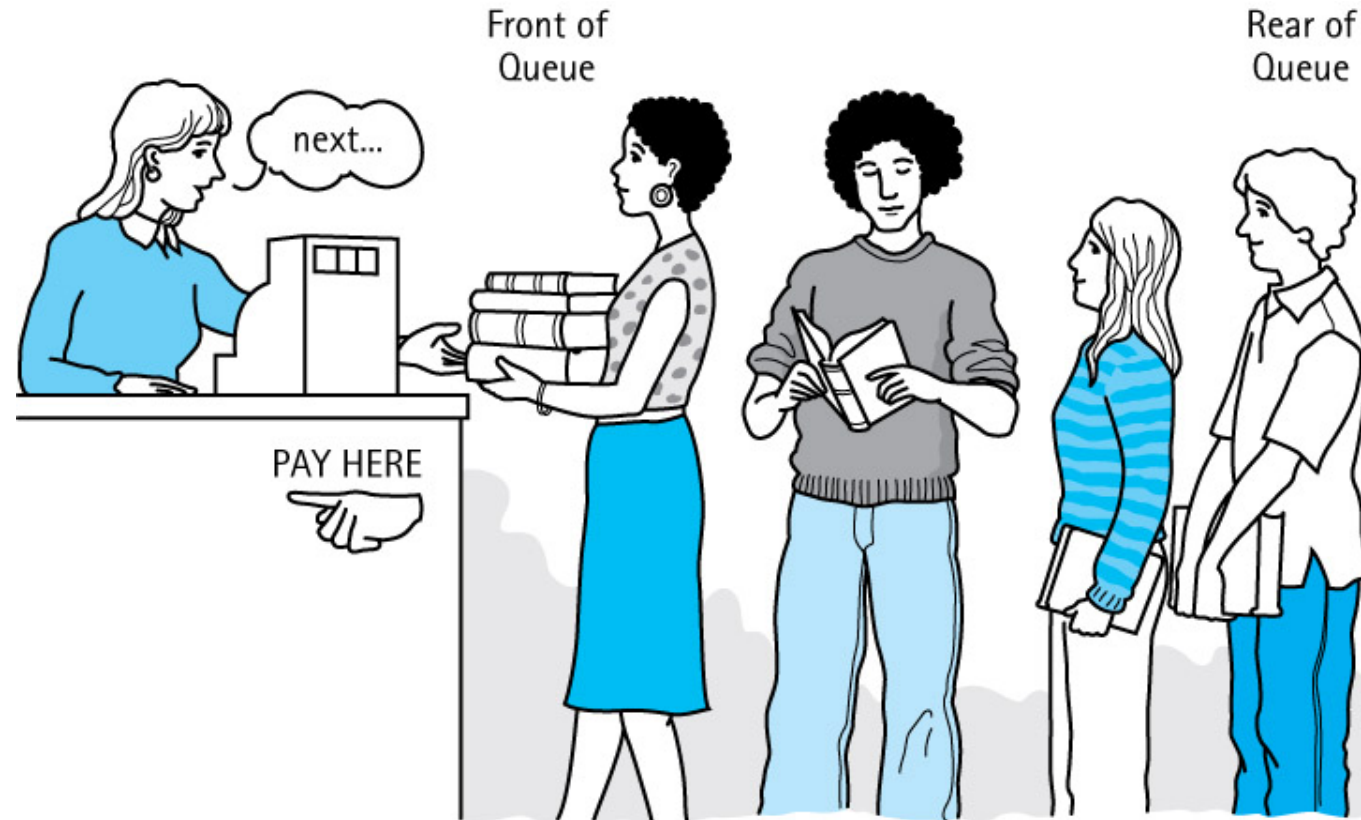
The Priority Queue



Queues are **F.I.F.O.** lists.
First In First Out

Queues

- **Queue** A structure in which elements are added to the rear and removed from the front; a “first in, first out” (FIFO) structure



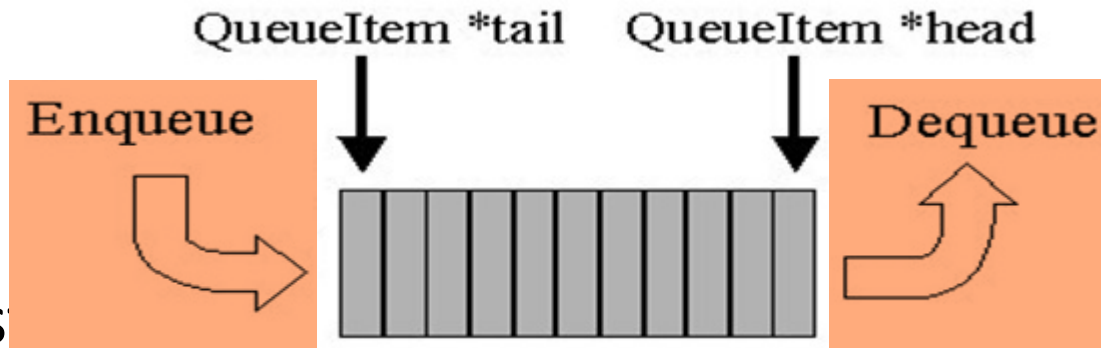
The Queue ADT

- Queues are lists
- Two main operations:
 - Enqueue
 - Dequeue

- Queue is: FIFO list

- Implementation

- [Array](#)
- [Linked-List](#)



Effects of Queue Operations

Originally

Queue is empty

enqueue block2



front = block2

rear = block2

enqueue block3



front = block2

rear = block3

enqueue block5



front = block2

rear = block5

dequeue



front = block3

rear = block5

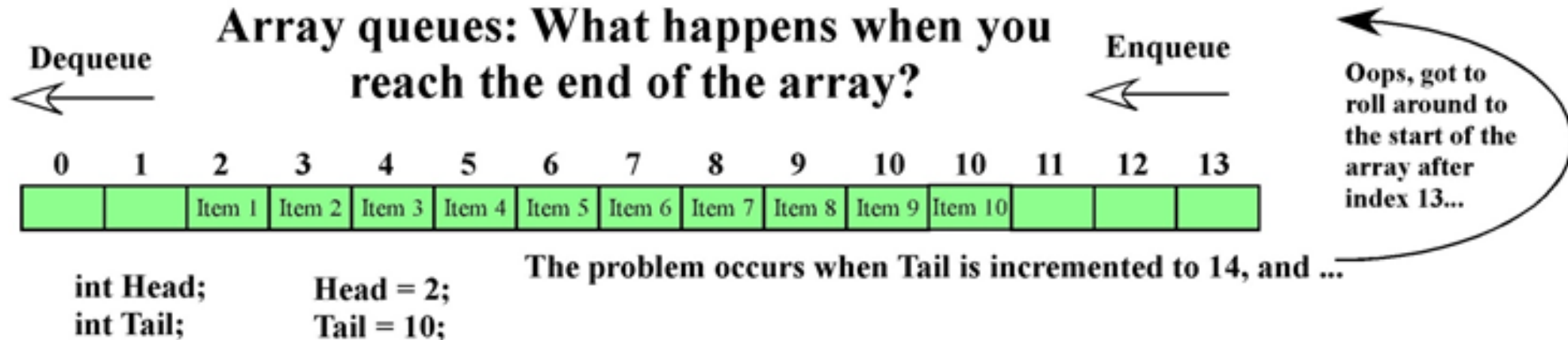
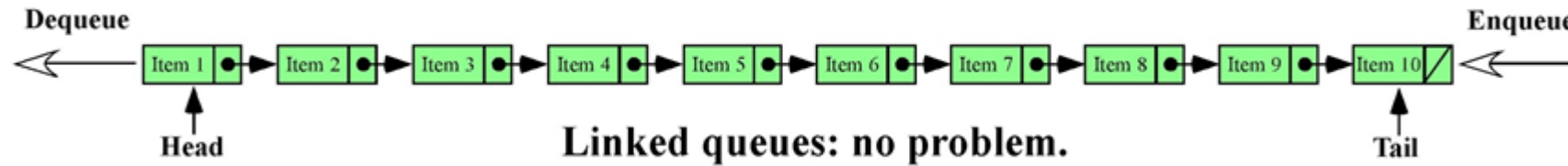
enqueue block4



front = block3

rear = block4

Queue

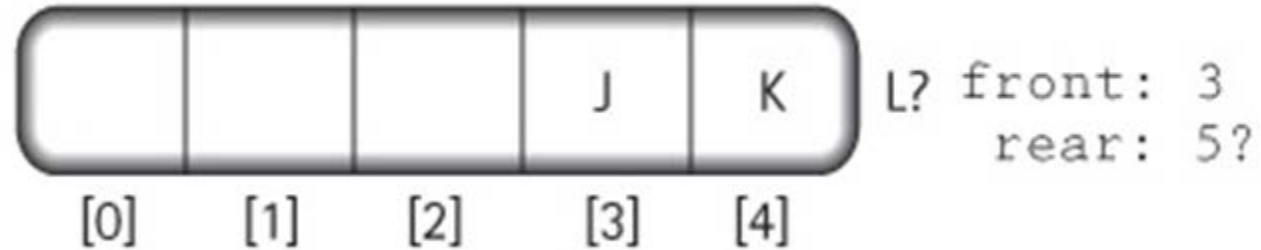


Queue Operations

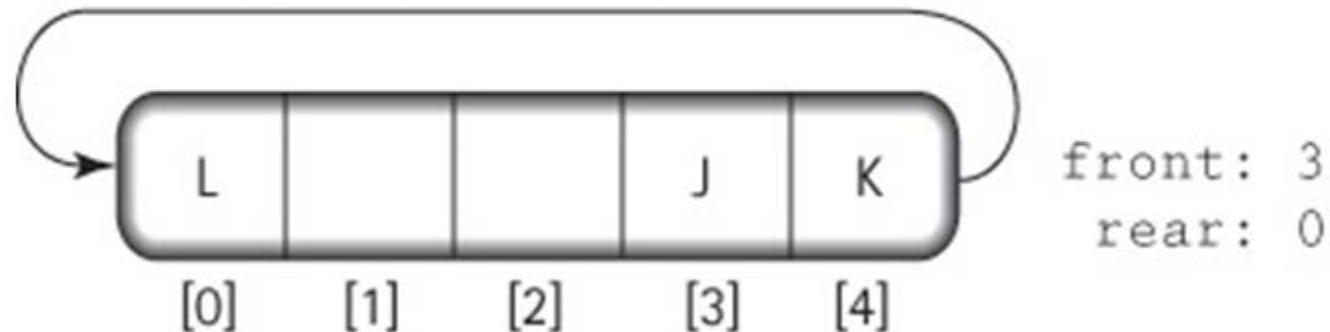
- To enqueue an element x,
 - we increment currentSize and tail,
 - then set theArray[tail]=x.
- To dequeue an element,
 - we set the return value to theArray[head],
 - decrement currentSize, and
 - then increment head.
- checking for errors
 - whenever head or tail gets to the end of the array?
 - it is wrapped around to the beginning.
 - This is known as a **circular array** implementation

Wrap Around Solution

(a) There is no room at the end of the array



(b) Using the array as a circular structure, we can wrap the queue around to the beginning of the array



Comparing Queue Implementations

- Memory usage
 - Array-based:
 - takes the same amount of memory,
 - No matter how many slots use, proportional to current capacity
 - Link-based:
 - takes space proportional to actual size of the queue
 - each element requires more space than array
- Operation efficiency
 - All operations, for each approach, are $O(1)$
 - Except for the Constructors:
 - Array-based: $O(N)$
 - Link-based: $O(1)$

Applications using Queue

- Several of these uses are found in
 - **Graph** theory
 - When jobs are submitted to a printer,
 - they are arranged in order of arrival.
 - jobs sent to a line printer are placed on a queue.
 - Virtually every real-life line is (supposed to be) a queue.
 - For instance, lines at ticket counters are queues,
 - service is first-come first-served.
 - computer networks
 - the disk is attached to one machine, known as the **file server**.
 - Users on other machines are given access to files
 - on a first-come first-served basis