Lecture 15: Queues and Practice Problems

CS 62

Spring 2018
Alexandra Papoutsaki & William Devanny
Implementing Queue using an array

• Assume we know max capacity required

• Keep track of **head** and **count** and have the data wrap around at the end of the array
  • Always add at the \( \text{tail} = (\text{head} + \text{count}) \mod \text{data.length} \)
  • Always remove from the **head**

• Use modular arithmetic to calculate indices
  • e.g., \( \text{headIndex} = (\text{headIndex} + 1) \mod \text{data.length} \) when remove()

• [http://www.cs.pomona.edu/classes/cs062/structure5/QueueArray.java](http://www.cs.pomona.edu/classes/cs062/structure5/QueueArray.java)
Practice Time

4 classic interview problems on linked lists, queues, and stacks

Work in groups

Pick 2 problems (one from linked lists and one from queues & stacks)

Write unit tests

Work on both for the next ~30'
Continue with a new one if done before the allotted time

Assume you can use data structures offered in structure5 package
Write a Java program that:

1) Removes duplicate nodes in an unsorted singly linked list
   • Hint: Remember that you can use two pointers to traverse a list

2) Returns the kth to last element of a singly linked list
   • Hint: Think recursion

3) Represents a queue using two stacks. Should support enqueue, dequeue, peek, size

4) Reverses a queue using a stack