

Slide 1

### Administrivia

- Homework 4 on Web. Design due Tuesday (10/16), code Thursday (10/18).

Slide 2

### Recap — ADTs, Array-Based Stacks and Queues

- Abstract data type — define possible “values”, list of operations. Could capture as Java `interface`.
- Stack ADT — LIFO queue.
- Queue ADT — FIFO queue.
- Array-based implementations:
  - Stacks easy, queues somewhat trickier (“circular queue”).
  - General approach — decide what variables we need, what they should “mean”.
  - Error checking — Java-esque way is with exceptions.
  - Include `main` method for simple error checking.

## Lists

Slide 3

- List ADT:
    - “Values” are lists of elements.
    - Many operations possible — add element, remove element, search for element, etc., etc.  
(Also “walk through elements” with “iterator” — next time.)
  - Implementation:
    - Using an array.
    - Using a “linked list”.
- How do these compare with regard to efficiency of various operations?  
efficiency of memory use?

## Homework 4 Overview

Slide 4

- Start writing code for your game entities. Similar to what you did for player last time.
- Review/revise how you’re creating layout for your game. Several options.
- Write replacement for framework `GameEntityList`. This will be a linked list, based on discussion today and next time. You may find it helpful to draw pictures.

## Linked Lists

Slide 5

- Idea of linked lists is fairly simple — chain of “nodes”, each linked to the next (singly-linked list) or to the next and the previous (doubly-linked list).
- How to implement this in Java? First, draw pictures.
- Then think about what you need to turn the pictures into code. Probably you’ll need:
  - Variables (e.g., something to point to the first “node” (little box).
  - Classes-within-the-class (for nodes / little boxes, later for iterators).
  - Methods for interface.
- (Sketch the start of an implementation on the board.)

## Minute Essay

Slide 6

- What are you finding interesting / useful / difficult / annoying about the homeworks so far?