Updated readings on “lecture topics and assignments” page.

Reminder: Homework 3 design due 11:59pm today. Homework 3 code due Thursday.

Reminder: Quiz 3 Thursday. Likely topic is something involving arrays, maybe in the context of writing a class.

Midterm next Tuesday. I will post a review sheet on the Web before Thursday, and we can spend a few minutes in class talking about it.

---

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

- Abstract data type — define possible “values”, list of operations. Could capture as Java interface.
- Stack ADT — FIFO queue.
- Queue ADT — LIFO queue.
- Implementations based on Vector last time:
  - Stacks easy, queues somewhat trickier (“circular queue”).
  - General approach — decide what variables we need, what they should “mean”.
  - Include main method for simple error checking.
- Minute essay question: If we start with an empty stack and do push(5), push(7), pop(), push(2) — what’s the result?
General Homework Hints

- “Links” — a way to teleport player or other entity from one spot to another. Make this work via methods of Block and GameEntity interfaces.

- Some helpful tools — “screen editor” for laying out screens, “image editor” for drawing images. Or can use ImageIO.read() and files created by other programs. Appears that JPEG, bitmap, GIF, and PNG are supported.

Homework 3 Hints

- Player defines some constants you should use.

- You will implement KeyListener or one/both of the mouse-listener interfaces. When you do this, the framework will deliver key and/or mouse “events” to you.

- Most logic will go in update, getUpdateTime, and the listener methods.

- Think about what variables you will need.
More About Arrays and Sorting

- We looked briefly at the `Arrays` class and its methods. Let's look a little more at how we could use its sort methods to do (1) case-insensitive sort and (2) sorting of objects of a class we wrote . . .
- We could also code up a framework for comparing various sorts, by counting numbers of comparisons . . .

Lists

- List ADT:
  - “Values” are lists of elements.
  - Many operations possible — add element, remove element, search for element, etc., etc.
- Implementation:
  - Using an array.
  - Using a “linked list”.
Minute Essay

- Tell me about your experiences doing the homeworks so far: What has been difficult? easy? interesting? valuable? not valuable?