CS 232 Lab 4 September 30, 2016

Due: October 7, 2016, before 11am

Download the

Part 1: ArrayQueue does what?

An ArrayQueue is a queue implemented on top of an array (as opposed to using linked nodes). It works by starting at ary[0] for the front and back, putting new elements in the ary[back] location, and then incrementing back. Items are removed by taking them from ary[front], and then incrementing front. When the queue runs out of room at the end of the array (back = ary.length-1), if there is room at ary[0] then the back starts over at 0. (If there is not room at the front then the the array could be resized to make room, but our version doesn't do that.) Given a small queue and the following operations, it would go something like this:

new ArrayQueue<Integer>() a = new ArrayQueue<Integer>(3);

front = 0, back = 0, size = 0

a.enqueue(5);

5		

Queue: 5 front = 0, back = 1, size = 1

a.enqueue(6);

5	6	

Queue 5 6 front = 0, back = 2, size = 2

a.dequeue();

	6	

Queue: 6 front = 1, back = 2, size = 1

a.enqueue(7);

	6	7

Queue: 6 7 front = 1, back = 0, size = 2

a.enqueue(8);

8	6	7

Queue: 6 7 8 front = 1, back = 1, size = 3

a.dequeue();

8		7

Queue: 7 8 front = 2, back = 1, size = 2

Take a look at the driver provided in the ArrayQueue.zip file. Do not run the code at this point. Based on the examples above, walk through the code and figure out what the state of the queue and the ints **should** be at each step. Write it out on paper or use a text editor. You don't have to turn it in, but I won't help you with the code bits unless you've done this part. If you are feeling brave, you can take your chances without it. :-)

Part 2: This one's broken!

Okay, **now** you can run the code. Is the output different from what you wrote down for part 1? It should be! This code has a few errors in it, and it is up to you to fix it. Note that the toString method prints the queue in the order that the items should be in, not the order that they appear in the array. You might find jGrasp useful here.

Find the errors in the code and fix them. Add comments to the code where you find the errors: explain what the problem was and how you fixed it.

Bundle up your code and turn in the .zip file on Moodle!