Learning Outcomes

Upon completion of this lab, you should be able to do the following in Eclipse:

• Understand the circular array implementation of a queue.

General Lab Instructions to Help Labs Run Smoothly

• Read through the lab instructions before coming to the lab.
• Do the pre-lab preparation.

Preparation. Review the lecture notes of queues. Make sure you understand what a queue looks like when implemented as a linked list, an array, and a circular array.

IMPORTANT: Make sure you attend your lab session and sign an attendance sheet, show the result(s) of each exercise to your TA, and submit your .java files through OWL by 11:55pm on the same day as your lab session to get your marks for the lab.

Exercise 1: Completing the CircularArrayQueue Implementation

• Download CircularArrayQueue.java, QueueADT.java and EmptyCollectionException.java from the sample code page and fill in the missing code in the methods. The code for the helper method expandCapacity() is given; note that it is different from the expandCapacity() of the regular array implementations that we have seen for the Stack ADT and Queue ADT.
  – first() is similar to dequeue(), but doesn’t remove the element
  – isEmpty() and size() are trivial
  – for toString(), setting up the string is similar in concept to the toString() method in ArrayStack.java, which was similar to the following:

```java
1 public String toString() {
2     String result = "";
3     for (int i = 0; i < top; i++) {
4         result = result + stack[i].toString() + "\n";
5     }
6     return result;
7 }
```

Of course the attributes are no longer called stack and top, so you need to change those. But, more importantly, you will not be starting the array index at 0. The counter for the loop can still go from 0 to count, since you will still go through the for loop as many times as there are items in the queue. But the array index
will need to be a different variable that starts at the value in the attribute front. You will need to increment the array index in the body of the for loop, taking into account that it may need to wrap around to the beginning of the array again. It is the same idea as how rear is incremented in the enqueue() method and how front is incremented in the provided expandCapacity() method.

Exercise 2: Testing CircularArrayQueue

- Download the given test driver TestCAQ.java to test your implementation of the CircularArrayQueue class. This test driver is provided as an example of a test program that does thorough testing, in a way that displays whether or not the test results are the expected results.
- Answer the questions below before running TestCAQ on your circular array queue implementation, to make sure you understand how it works. Make corrections in your CircularArrayQueue.java until all the tests have been completed successfully.
  - In the main() method, what is the purpose of the second parameter in all the calls to the testing methods, for example the first 3 calls:
    1  t_isEmpty(first, true);
    2  t_size(first, 0);
    3  t_toString(first, "");
  - Is it possible to enqueue objects of different types to the queue first, for example:
    1  first.enqueue(new Integer());
    2  first.enqueue(new Float(i*10));
  - What is the type of the objects being enqueued in the following statements?
    1  first.enqueue("A");
    2  first.enqueue("S");