CSCI 241 Data Structures
Winter 2018
Laboratory Exercise 5

Objective

Practice in exception handling.

Getting Credit for Your Work

For this exercise, you need to submit your Java files through the Canvas web site within a zip folder named `lastname_firstname_lab5.zip`.

Your Task

The provided file MyQueue.java implements a queue as a finite circular array. The enqueue() method adds a DataItem to the tail of the queue and the dequeue() method removes a DataItem from the head of the queue. Method enqueue() returns a Boolean value: true, if the operation was successful, false if the queue was already full and the DataItem therefore could not be added to the queue. Method dequeue() returns a DataItem, or null, if the queue is empty.

Your task is to modify MyQueue.java so that enqueue() throws an exception on any attempt to add to a full queue and dequeue() throws an exception on any attempt to remove an item from an empty queue. The three control flow statements within the main() method need to be modified to use try … catch statements. The queue operations are attempted in the try-part and exception-handling is done in the catch part.

1. Any method that can throw an exception has its heading modified, for example:

   ```
   public void doStuff (int a, String s) throws exception {
   ```

2. A method throws an exception when things go wrong, for example:

   ```
   if (badThings)
      throw new Exception("Something really bad just happened");
   ```

3. Any method that calls another method which can throw an exception must itself either handle or throw the exception. To handle an exception, use a try … catch statement, for example:

   ```
   try {
      doStuff(3, "elephant");
   } catch (Exception e)
      System.out.println(e);
   }
   ```