Name: ____

CSCE 146 Program3 (Ch7) **Java Program: Queue**

Setup

1. Existing interfaces or abstract classes will not be implemented for this assignment
2. Use of code that implements a list from the program1 assignment or the Ch3 code on Blackboard

Program

1. **Queue Methods.** Your queue should have the following methods (handle out of bound index inputs) (Ch7 slides):
    - a. Queue() //default constructor, implement with linked list, size=0, fixCapacity = false
    - b. Queue(int capacity) //constructor for fix capacity queues, implement with an array, size=0, , fixCapacity = true
    - c. enqueue(int element) //add element at rear of queue and increase size of the queue by 1
        - i. if size = capacity and fixCapacity = true then don't add element and tell the user already at capacity
        - ii. if size = capacity and fixCapacity = false then add element and increment capacity
    - d. dequeue() // return element at the front of the queue, remove front element, and reduce size of queue by 1
        - i. if size = 0 then you can't return element, tell user queue is empty
    - e. peek() //return element at the front of the stack, do not change queue or size
    - f. capacity() //returns the capacity of the queue, also prints if this is fixed or not
    - g. isEmpthy() //returns true if size = 0, else return false
    - h. isFull() //returns true if size = capacity and fixCapacity = true, else return false
    - i. size()//return the size of the queue, the number of elements actually in the queue
2. **Default (no limit capacity) Queue Implementation.** Implement your Queue by using a list:
    - a. Decide what corresponds to front and rear
    - b. Decide which list methods map to the queue methods above
    - c. Have methods in part 1 above call methods in list to get results
3. **Limited Capacity Queue Implementation.** Implement your Queue by using an array:
    - a. Decide what corresponds to front and rear
    - b. Follow the guidelines in the book and slides (ch7) by wrapping your array
    - c. Create methods for your array that match the queue methods above
4. **User Interface.** Provide way to access and test all the methods above (such as scanner)
    - a. Be sure to initially give them the option of a limited capacity queue or not

Be sure to:

1. Use proper and clear comments and variable names
    - a. Include the following information at the top of your code:

      Name: *Your Name*
      Assignment: Program3
      Class: CSCE 146
      Semester: Spring 2014
      School: USC Sumter

Turn in:	demonstrate your program to the instructor

Grading:

Function:	
Objectives met of part 1	20%
Objectives met of part 2	20%
Objectives met of part 3	20%
Objectives met of part 4	20%
Organization: readable code and clear documentation	10%
Interface: user experience (clear direction) and handling of invalid input	10%