CPSC 2151

Lab 5

Due: Friday September 28th at 11:59 pm

In this lab you will be working with interfaces to add additional functionality to our IntegerQueue interface. You will be building off the code used in Lab 4.

Instructions

1. Override the ToString method for both of your existing implementations.
    a. Note: You cannot do this in a default implementation due to issues with multiple inheritance
2. Add the following methods to your IntegerQueueI interface using Default Implementations. Include all contracts and Javadoc comments for these methods.
    a. Integer peek() – returns the Integer at the front of the Queue, but does not remove it from the Queue.
    b. Integer endOfQueue() – returns the Integer at the end of the Queue, but does not remove it from the Queue
    c. Void insert(Integer x, int pos) inserts x at position pos in the Queue. Pos index starts at 1, so the item at the very front of the queue is pos 1.
    d. Integer remove(int pos) – removes whatever Integer was in position pos in the Queue and returns it. Pos index starts at 1, so the item at the very front of the queue is pos 1.
    e. Integer get(int pos) - – returns whatever Integer was in position pos in the Queue and without removing it. Pos index starts at 1, so the item at the very front of the queue is pos 1.
3. Create a class file called IntegerQueueApp and copy the following code into it:

```
private static Scanner in;

    public static void main(String[] args) {

        in = new Scanner(System.in);
        System.out.println("Enter 1 for array implementation or 2
for List implementation");
        int answer = in.nextInt();

        IntegerQueueI q;

        if(answer == 1) {
            q = new ArrayQueueImp();
        }
        else
        {
            q = new ListQueueImp();
        }
```

```
        displayMenu();
        answer = in.nextInt();
        while(answer != 8)
        {
            if(answer == 1)
            {
                addToQueue(q);
            }
            else if(answer == 2)
            {
                getNext(q);
            }
            else if(answer == 3)
            {
                peekInQueue(q);
            }
            else if(answer == 4)
            {
                peekAtEnd(q);
            }
            else if(answer == 5)
            {
                insertInQueue(q);
            }
            else if(answer == 6)
            {
                getFromQueue(q);
            }
            else if(answer == 7)
            {
                removeFromQueue(q);
            }
            else
            {
                System.out.println("Not a valid option!");
            }
            System.out.println("Queue is: ");
            System.out.println(q.toString());
            System.out.println(" ");
            displayMenu();
            answer = in.nextInt();
        }



    }

    private static void displayMenu()
```

```
    {
        System.out.println("Select an option: ");
        System.out.println("1. Add to the Queue");
        System.out.println("2. Get next number from the Queue");
        System.out.println("3. Peek at the front of the Queue");
        System.out.println("4. Peek at the end of the Queue");
        System.out.println("5. Insert in the Queue");
        System.out.println("6. Get a position in the Queue");
        System.out.println("7. Remove from a position in the
Queue");
        System.out.println("8. Quit");

    }
```

4. Complete the methods called in the main function to complete the specified tasks. You should not edit the main function.
    a. Make sure all requires clauses for your IntegerQueue methods are met. You do not need to add contracts for the methods in the IntegerQueueApp class. These methods will need to be private static and void.
    b. Note that each method has the Queue passed in as input. You should not make this a variable of the QueueApp class to avoid this.
    c. See my sample input at the end of the instructions for an example of how the program should work.

Before Submitting

You need to make sure your code will run on Unix. In order to do that you will first need to log onto the unix machine and create a new directory. Our directory has to match the package we created for our assignment.

Submitting your file

You will submit all your files using handin in the lab section you are enrolled in. If you are unfamiliar with handin, more information is available at https://handin.cs.clemson.edu/help/students/

Groups

You may, but are not required to, work with a partner on this lab. Your partner must be in the same lab section as you, not just the same lecture section. If you work with a partner, only one person should submit the assignment. You should put the names of both partners in a comment at the top of the file in order for both partners to get credit. This assignment may take more than just the lab time. Make sure you are able to meet outside of class to work on the assignment before you decide to work with someone else. Remember to actively collaborate and communicate with your partner. Trying to just divide up the work evenly will be problematic.

Lab Report:

Along with your code you must submit a well formatted report with the following parts:

Requirements Analysis

Fully analyze the requirements of this program. Express all functional requirements as user stories. Remember to list all non-functional requirements of the program as well.

Design

Create one diagram that has the IntegerQueueI interface and both implementations. They should be connected to show the implements relationship (shown with an arrow, like inheritance is). Remember that the class diagrams for the implementations should not include anything that is inherited from the interface. Note: You created a similar diagram last week that needs updating.

These diagrams should be created electronically so they are easy to read. I recommend draw.io

Testing

Describe how you tested your program to ensure it was functioning correctly. We have not covered testing in detail this semester, so the requirements at this point are pretty relaxed. Just describe your testing process. Include what specific inputs you used, and how you expected the program to react to those inputs. Make sure to test your program thoroughly.

Deployment

Create a makefile that will handle the compilation and running of the program. In your report you should include instructions on how to use the makefile, and the makefile should be included in the directory with the code.

Your lab report should be one file, and converted to a PDF

Sample Input/Output

Note: This sample input does not include all possible error messages (such as the Queue being full)

```
Enter 1 for array implementation or 2 for List implementation
1
Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
2
Queue is empty!
Queue is:


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
```

```
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
3
Queue is empty!
Queue is:


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
4
Queue is empty!
Queue is:


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
6
Queue is empty!
Queue is:


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
7
Queue is empty!
Queue is:
```

```
Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
1
What number to add to the Queue?
42
Queue is:
42


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
1
What number to add to the Queue?
37
Queue is:
42
37


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
1
What number to add to the Queue?
36
Queue is:
```

```
42
37
36


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
2
Next number is 42
Queue is:
37
36


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
1
What number to add to the Queue?
42
Queue is:
37
36
42


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
```

```
3
Peek: 37
Queue is:
37
36
42


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
4
Peek at end: 42
Queue is:
37
36
42


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
6
What position to get from the Queue?
7
Not a valid position in the Queue!
What position to get from the Queue?
2
36 is at position 2 in the queue
Queue is:
37
36
42


Select an option:
```

```
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
5
What number to add to the Queue?
17
What position to insert in?
9
Not a valid position in the Queue!
What position to insert in?
1
Queue is:
17
37
36
42


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
5
What number to add to the Queue?
19
What position to insert in?
5
Queue is:
17
37
36
42
19


Select an option:
1. Add to the Queue
2. Get next number from the Queue
```

```
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
7
What position to remove from the Queue?
1
17 was at position 1 in the queue
Queue is:
37
36
42
19


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
7
What position to remove from the Queue?
5
Not a valid position in the Queue!
What position to remove from the Queue?
4
19 was at position 4 in the queue
Queue is:
37
36
42


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
```

```
7
What position to remove from the Queue?
2
36 was at position 2 in the queue
Queue is:
37
42


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
9
Not a valid option!
Queue is:
37
42


Select an option:
1. Add to the Queue
2. Get next number from the Queue
3. Peek at the front of the Queue
4. Peek at the end of the Queue
5. Insert in the Queue
6. Get a position in the Queue
7. Remove from a position in the Queue
8. Quit
8

Process finished with exit code 0
```