CSCI 2320
Spring 2017
Exam 3

Last Name: _____

First Name: _____

Write your answers on the answer sheets provided. You may refer to printouts of any source code you have written.

```
class ListNode {
   int data;
   ListNode next;
}
class MyList implements Comparable<MyList> {
   private ListNode head;
   public MyList() { head= null; }
   public void add(int elem) { ... /* adds to front of list */ }
   public void remove(int elem) { ... /* removes specified element list */ }
   public int size() { ... /* returns number of elements in the list */ }
}
class MyStack<T> {
   // internal implementation not specified, but does have these methods
   public void push(T elem) {}
   public T pop() {}
   public boolean isEmpty() {}
}
// A FIFO queue build on a linked list (and with an unusual property)
class MyQueue<T> {
   // internal implementation not specified, but does have these methods
   public void insert(T elem) {}
   public T remove() {}
   public int size() {}
}
```

1. (4 pts) Expand the acronym LIFO. What data structure is it associated with?
2. Answer these questions about binary search:
   (a) (3 pts) Why can't binary search be applied to every array?
   (b) (2 pts) About how many steps/iterations would it take a binary search to find an element in a list of 1000 elements.
3. (6 pts) Show how the following array would be processed if it were partitioned one time using the partition algorithm that accompanies Quicksort.

   $< 44, 55, 120, 20, 112, 50, 63, 17, 30, 60, 80 >$
4. Answer these questions about linked lists and the `MyList` class given above.

(a) (6 pts) Write a `count(int elem)` method for the `MyList` class that will return the number of elements in the `MyList` object that match the parameter `elem`.

(b) (10 pts) The notation in the header of `MyList` that says:

```
implements Comparable<MyList>
```

is unusual for a linked lists but is certainly possible. Write a method that would have to be added to the `MyList` class in order to support the claim made by this notation.

We define two lists to be equal if they have the same number of elements and if those elements have the same value and order. If one list has fewer elements than the other then it is considered less than the other. If two lists have the same number of elements then find the first data value that is different and use that element's value to determine which of the lists is smaller than the other.

(c) (6 pts) Write a section of code that will create two `MyList` objects and will insert 3 integer values into each one. Then utilize your method from problem 4b to display "same", "less than", or "greater than" if the first list is the same, less than, or greater than the second list, respectively.

(d) (12 pts) Write a `merge(MyList other)` method for the `MyList` that will combine the contents of the current list with the list specified by the parameter `other`. The current list should be modified but the `other` list should not. When combining list elements interleave them starting with the current list.

5. Answer these questions about stacks and the `MyStack` class given above.

(a) (4 pts) Suppose that when a letter is encountered we push it on the stack and when an asterisk is encountered we pop the stack. Show the sequence in which letters would be popped for the following string:

```
ST*ELEH**EBOL***TS***UR*R*****
```

(b) (4 pts) Contrast an array-based stack implementation with a list-based stack implementation. Give advantages of each over the other.

(c) (10 pts) Write a method called `convertToDecimal(String)` that accepts a parameter string containing zeroes and ones. You will use a stack to calculate the base-10 integer corresponding to the specified binary number. Suppose the string parameter is "10010". The decimal equivalent of this value can derived by multiplying each digit by consecutively larger powers of two starting with the least significant digit. So,

$$\begin{aligned} 10010 &= 0 \times 1 + 1 \times 2 + 0 \times 4 + 0 \times 8 + 1 \times 16 \\ &= 0 + 2 + 0 + 0 + 16 \\ &= 18 \end{aligned}$$

To do this you will want to process the characters of the string in reverse order which you will accomplish by putting the individual characters on a stack and then popping them off to do the calculations. Your method should return an

integer value. You may assume the existence of a working, generic `MyStack` class shown at the beginning of the exam.

6. Answer these questions about queues.

   (a) (4 pts) Suppose that when a letter is encountered we put it in the queue and when an asterisk is encountered we remove from the queue. Show the sequence in which letters would be removed for the following string:

   ```
   ST*ELEH**EBOL***TS***UR*R*****
   ```

   (b) Imagine you are wanting to create a program that simulates lines at a fast food restaurant and that you will use the provided `MyQueue` class which includes three working methods: insert, remove, and size. You will store in the queues `Person` objects as defined here:

   ```
   public class Person
   {
      private String name;
      private int enterTime;
      private int leaveTime;
      public Person(String name) {
         this.name= name;
         enterTime= 0;
         leaveTime= 0;
      }
      public void setLeaveTime(int lt) {
         leaveTime= lt;
      }
      public String toString() {
         return name+" ("+(leaveTime-enterTime)+")";
      }
   }
   ```

   Suppose a large fast-food restaurant has 10 cash registers. Write a section of code to do the following:

   i. (6 pts) Create an array of 10 queues of `Person` objects.

   ii. (4 pts) Use a loop to insert one person in each queue. The individuals should be named "Person 1", "Person 2", etc.

   iii. (6 pts) Write a method called `showLines` that accepts one parameter: an array of `MyQueue<Person>` objects. The method will, for each queue in the array, display the index of the queue and the number of people waiting in the queue.

   iv. (4 pts) Write a section of code that will remove a person from the first queue in the array and will set their leave time to 25. Then display the person's name along with their total wait time in parentheses. NOTE: Wait time is the time they entered subtracted from the time they left.

7. (4 pts) Which homework assignment in this course was your favorite? Why?

8. (4 pts) Which topic in this course was your favorite?