Programming Assignment #2

75 points

Due Date/Time:

Your project is due on Monday March 12th at 8 A.M. Hardcopy of your Java source code files is also due at that same time.

The Program:

For this assignment, you will write two more implementations of a **`Priority Queue`**. Using the same interface as program #1, you will write two linked listed implementations.

Your implementations will be:

1. Ordered Singly Linked List
2. Unordered Singly Linked List

Both implementations must have identical behavior, and must implement the PriorityQueue interface used in program #1. The implementations must have a single no-argument constructor. As linked lists are never full, there is no default size or maximum size.

Thus, your project will consist of the following files. You must use exactly these filenames.

- `PriorityQueue.java` The ADT interface (provided in Assignmeng 1)
- `OrderedLinkedListPriorityQueue.java` The ordered list implementation.
- `UnorderedLinkedListPriorityQueue.java` The unordered list implementation.

Additional Details:

- Each method must be as efficient as possible. That is, a O(n) is unacceptable if the method could be written with O(log n) complexity.
- You may not make any modifications to the PriorityQueue interface provided. I will grade your project with my copy of this file. This interface is UNCHANGED from project #1

- All relevant requirements from the first assignment apply here. Of course, arrays will have a limited size but linked lists do not. Thus, all references to maximum size are not relevant for this assignment.
- The `isFull()` method should be hardcoded to return false.
- You should ignore (but do not remove) the DEFAULT_MAX_CAPACITY variable in the interface.

Turning in your project:

To submit your project, you must copy both Java source code files into your `handin/prog2` subdirectory. You will submit a printout of these files in class on the due date.*[**IMPORTANT NOTE:** Do not recreate the data_structures subdirectory in the handin subdirectory--just copy your two files into the `handin/prog2/` directory itself.]* Be sure to check the Program Submission Guidelines page.

Cheating Policy

There is a zero tolerance policy on cheating in this course. You are expected to complete all programming assignments on your own. Collaboration with other students in the course is not permitted. You may discuss ideas or solutions in general terms with other students, but you must not exchange code. During the grading process I will examine your code carefully. Anyone caught cheating on a programming assignment (or on an exam) will receive an "F" in the course, and a referral to Judicial Procedures.