Homework 5: Heaps

Due: *06/17/13 11:55*

Important

There are a few guidelines you must follow in this homework. If you fail to follow any of the following guidelines you will receive a 0 for the entire assignment.

- All submitted code must compile under JDK 7. This includes unused code, don't submit extra files that don't compile. Java is backwards compatible, so if it compiles under JDK 6 it should compile under JDK 7.
- DO NOT include any package declarations in your classes.
- DO NOT change any existing class headers, constructors, or method signatures. It is fine and encouraged to add extra methods and classes.
- DO NOT import anything that would trivialize the assignment. (e.g. don't import java.util.LinkedList for a Linked List assignment. Ask if you are unsure.)
- You must submit your source code (ie. the .java files), NOT the compiled bytecode (ie. the .class files)

After you submit your files, redownload them and run them to make sure they are what you intended to submit. We are not responsible if you submit the wrong files. DO NOT submit .class files.

Binary Heap

A binary heap is a heap data structure using a binary tree. There are two properties to a binary heap:

- *Complete Tree*: All levels of the binary tree, except for possibly the deepest level, are completely filled, and the last level (if present) is filled from left to right.
- *Heap Property*: Each node is greater (less than depending on the heap) or equal to each of its children. A heap with the smallest item being the root is called a min-heap while a heap with the largest item being the root is called a max-heap.

Insert

Data inserted to the binary tree will be inserted to maintain the property of a Complete Tree. To maintain the heap property, the added data will move up the tree as far as it can. This swapping up operation is called swim.

Remove

Removing data will only remove the root of the tree. To maintain the property of a Complete Tree, the last element on the last level of the tree will replace the root. To maintain the heap property, the element will continue to swap with its lesser or greater of the two children depending on the type of heap. This swapping down operation is called sink.

Heapsort

Heapsort is a sorting algorithm that constantly uses the remove function of a heap to created a sorted list. First a heap is created from the given list. Then a sorted list is created back to front by repeatedly removing the root element from the heap and replacing it with the last element in the heap. Each time the size of the heap is decremented which prevents replacing the root with correctly placed elements.

Assignment

You will be implementing a max binary heap backed by an array. Your largest element will be stored at index 0 of the array. Do not worry about duplicate data. You will also be creating a method that performs heapsort on a COPY of the current heap which will return a list sorted in ascending order.

Deliverables

You must submit the following files:

- BinaryHeap.java

You may attach them individually or submit them in a .zip folder.