UNIVERSITY OF VIRGINIA, DEPARTMENT OF COMPUTER SCIENCE

Advanced Sorts - Implementation

Dr. Mark R. Floryan

November 5, 2019

1 SUMMARY

For this homework, you will be implementing two more sorting algorithms.

1. Download the provided starter code.
2. Implement two sorting algorithms.
3. Use the provided tester to check if each methods are working correctly.
4. **FILES TO DOWNLOAD:** AdvancedSorts.zip
5. **FILES TO SUBMIT:** AdvancedSorts.zip

1.1 ADVANCEDSORTS.JAVA

You will be implementing two sorting algorithms for this homework. They are listed below:

```
/**
 * Recursive merge sort and associated private helper method
 * the second method below provides the portion of the array
 * (i.e., index i to j inclusive) that we want to sort.
```

```
5     */
  public static<T extends Comparable<T>> void mergeSort(T[] list);
7   private static<T extends Comparable<T>> void
                                    mergeSort(T[] list, int i, int j);
9
  /**
11    * Recursive quick sort and associated private helper method
      * the second method below provides the portion of the array
13    * (i.e., index i to j inclusive) that we want to sort.
      */
15  public static<T extends Comparable<T>> void quickSort(T[] list);
  private static<T extends Comparable<T>> void
17                                  quickSort(T[] list, int i, int j);
```

You may add additional helper methods that might be useful in implementing these sorting algorithms. Some methods that may be useful, but are optional, include swapping two elements given indices i and j, merging two sorted lists into a single sorted list, partitioning a list around a pivot, etc.

The provided tester (main method) will call each of your sorting methods one at a time. Above the main method is a variable you can change to increase or decrease the size of the lists to sort while testing. The tester will call each of your two sorting methods one at a time. For each, it will take the result and check if 1) the sizes of the unsorted and sorted lists are still the same, 2) the elements in the sorted list are indeed in sorted order, and 3) the sorted list contains the same elements in total that the original list does. If any of these checks fail, you should get a notification to that effect.