Assignment Prefix: lab11
Points: 100
Due Date: Wednesday, November 23, 2016 @ 11:59pm

For this assignment you are going to implement several sorting algorithms.

RESTRICTIONS:

- You may **NOT** import **java.util.Comparator**
  - you must write your own comparators
- You may **NOT** import **java.util.Arrays**
  - You must write your own copy methods for any arrays
  - You may **NOT** use any of the Java Array sorting features.
  - **Exception, in the mergeSort <u>you can call</u> the Arrays.copyOfRange method**
- You may **NOT** import any other Java container class.
  - e.g. you must use your own Queue, LinkedQueue, SinglyLinkedList, etc. classes.

Task 1:

Create an employee Class that encapsulates the concept of an employee. The attributes of an employee are:

- id
  - a random integer in the range 0 to 99999999 (i.e. like a social security number)
  - we will ignore the fact that we may get duplicate id numbers
- name
  - a String of a random length between 5 and 10 characters (inclusive) made up of a random set of lower case characters
- dept
  - a random integer in the range 1 to 5 (inclusive)

- hired
    - a random integer in the range 1995 to 2010 (inclusive)

Task 2:

Create a class named Sort that will act as a container for the following generic array sorting algorithms:

- simpleBubbleSort
    - a brute force bubble sort that just uses a pair of nested loops
    - this needs to be a generic bubble sort
    - this needs to be a stable sort
- mergeSort
    - this should be the recursive mergeSort described in the textbook
- quickSort
    - this should be the recursive quickSort described in the textbook
    - you may have to modify this code
- radixSort
    - this should be a generic sort
    - the radixSort should be able to support between two and four keys
    - the first parameter in the parameter list should be the array being sorted.
    - the remaining parameters in the parameter list should be the keys, ordered left to right from most significant to least significant

Task 3:

- Create a client class that
    - Generates an array of any number of employees
    - Sort the employee array on name using the merge sort
    - Sort the employee array on dept using the quick sort
    - Sort the employee array using the radix sort so that
        - All employees are sorted by department
        - Within a department grouping all the employees are sorted by hire date
        - Within a department and hire date grouping all the employees are sorted by their name