LAB EXERCISE

Mergesort (Recursive)

Assignment:

1. This assignment should be done in your completed Lab 22.1. Modify the stubs which already exist in that lab.

2. Using the merge program in lab exercise L.A.24.1, `Merge`, as a starting point, write a recursive `mergeSort` method as described in the student outline. Pseudocode for the recursive `mergeSort` method is given below.

```
void mergeSort(int[] a, int first, int last)
//  Recursively divides a list in half, over and over. When the
//  sublist has one or two values, stop subdividing.
{
   if (sublist has only one value)
      do nothing
   else if (sublist has two values)
      sort it if necessary
   else    // recursion, divide list into two halves
      Find midpoint of current sublist
      Call mergeSort and process left sublist
      Call mergeSort and process right sublist
      merge left and right sublists
}
```

3. You will have to modify the `merge` method to fit the necessary calls of the `mergeSort` method.

Instructions:

1. After confirming that your mergesort works, paste the necessary routines into your sorting template program and count the number of steps for a recursive mergesort. Record the number of steps on the worksheet from Lesson 22, Worksheet W.A.22.1, *Comparison of Sorting Algorithms*.

2. Turn in your source code and a printed run output of 100 numbers, sized from 1-200. If possible, print only `merge` and `mergeSort` methods.