Indian Institute of technology, Guwahati

Department of Computer Science and Engineering

Data Structure Lab: (CS210)   **Mid Semester Examination**   **Total Marks: 25**

Date: 11th August, 2017

1. Write a function which will generate the array of n random elements for given input size n. Call below function with the randomly generated array. Count the number of comparisons. Report the number of comparisons. **[5]**

2. The merge sort is actually slower than simple $O(n^2)$ sorts for small input sizes. The following figure was created by timing merge sort, insertion sort, and binary insertion sort on small randomly ordered lists from size 2 to size 64. As you can see, binary insertion sort is the fastest of the three algorithms until around n = 55. At that point, merge sort becomes faster and it remains faster for all larger inputs. **[20]**
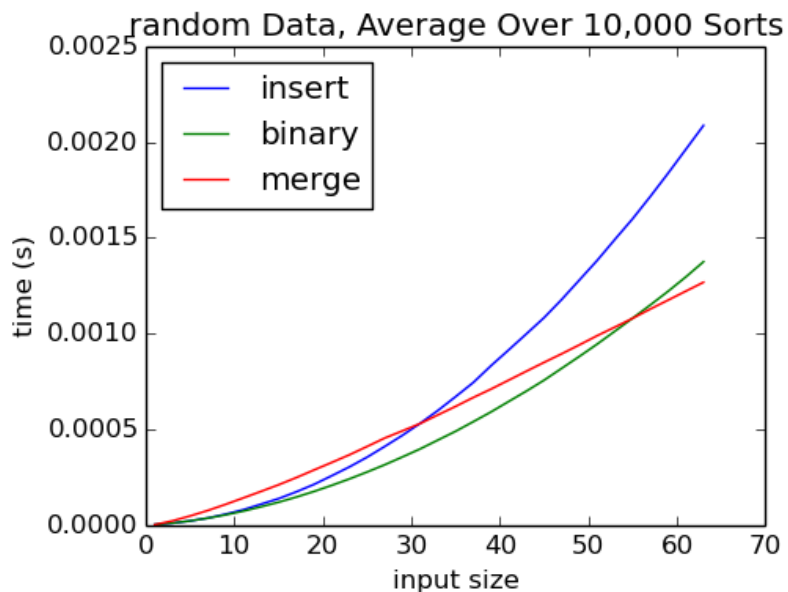
As reminder, the following pseudocode describes the overall logic of the merge sort Algorithm:

merge_sort(sub-list)

   If sub-list is has more than one entry:

      Recursively merge_sort the left half

Recursively merge_sort the right half

Merge the two sorted halves.

This logic recursively splits the original list into smaller and smaller sub-lists until the recursion bottoms out at lists of size one. This means that every time a large list is sorted, there are many recursive calls to merge sort that have small input sizes. In light of the figure above, that approach doesn't make much sense: merge sort is not a competitive sorting algorithm on small inputs. It would make more sense to recursively break the input into smaller and smaller pieces until some threshold is reached, and then switch strategies to a sorting algorithm that is more efficient on those small inputs.

The following pseudocode describes this alternate approach:

merge_sort(sub-list)

If sub-list is has fewer than MERGE_SORT_THRESHOLD entries:

Sort the sub-list with binary insertion sort.

Otherwise:

Recursively merge_sort the left half

Recursively merge_sort the right half

Merge the two sorted halves.

Assume MERGE_SORT_THRESHOLD is 55 as shown in the graph.

Implement the modified version of merge sort.