

CS 302
Assignment 5
Quicksort and Mergesort

In this assignment you need to write a program that implements the Quicksort and Mergesort Sorting algorithms.

You will prompt the user to input a series of characters and then sort the list using

- 1) Mergesort
- 2) Quicksort
- 3) A combination of both that will use Quicksort to sort until the sublist is smaller than 10 elements at which point it will use mergesort to sort the remaining items.

Your program must sort uppercase and lowercase letters alphabetically. Consider both, uppercase and lowercase letters to be of the same value. For example:

deAaCbbaFaBgZbabAbBzBbbeC

Would get sorted as:

aAaaaAbBbbBBbbbbCCdeeFgZz

So do not just use the ascii value. I suggest you implement a compare function that takes this into account and then just call that function in each of your sorting algorithms.

Use linux redirection to pass in the characters. I suggest you try a large number in order to have better analysis of the asymptotic complexity.

Copy the array with the characters you read into two other arrays so you have 3 copies of the list. So you can then run all three algorithms and output (to the terminal) the resulting sorted list.

Sample Input:

FBfvRaf

Sample Output:

MergeSort:
aBFffRv

Quicksort:
aBFffRv

Combination:
aBFffRv

Comparison Data (Feel Free to format this as you want, but show the sorting output first)

Reminder: DO NOT USE STL or any sort functions To complete assignment You have to code this yourself from scratch!

Please name your assignment: cs302hw5.cpp (source code) and cs302hw5.pdf (for report)

Notes:

- For your initial quicksort implementation I suggest you go with the pivot as the last element as shown in class.
- Document your source code and follow the online rubric!

Part II, Report.:

When completed, create and submit a write-up (PDF Format) of no more than three pages (1 page is fine, half a page is not):

- Name, Assignment, Section
- Summary of the implementation. Especially how you picked the pivoted and worked with the lo and hi indices.. Analyze and provide the Big-Oh Notation for each of the sorting algorithms. Add some count code as you did in previous assignments and provide a comparison between all 3 sorting algorithms (Merge, quick, and the combo) and find out which was the most efficient. Also play with how you pick the pivot and see how it affects the above. (select the first element and immediately swap it to the end)
- A discussion on the efficiency of the algorithm. Is there a way another way to write the algorithm to make the time complexity better?
- Notice that because mergesort is stable and quicksort is not. You will get a slightly different order in terms of uppercase and lower of the same letter. Does it work as you expected in regards to stable sorting algorithms?
- Try