

Homework 8 Programming

(Maximum 25 points)

Due 11:59 pm Wednesday April 7, 2021

Submit your Java codes in one file via Blackboard. Refer to the grading guidelines posted on Blackboard to understand how the submitted exercises will be graded.

1. (25) [Counting inversions: programming] Implement the sort-and-count algorithm discussed in class and shown below. (Recall that a pair of numbers in the input array L is said to be inverted if the number on the left is larger than the number on the right.) The algorithm of Merge-and-Count is not shown here, but we discussed it in class.

```
Sort-and-Count(L) {  
    if list L has one element  
        return 0 and the list L;  
    Divide the list into two halves A and B;  
    ( $r_A$ , A)  $\leftarrow$  Sort-and-Count(A);  
    ( $r_B$ , B)  $\leftarrow$  Sort-and-Count(B);  
    ( $r$ , L)  $\leftarrow$  Merge-and-Count(A, B);  
    return  $r = r_A + r_B + r$  and the sorted list L;  
}
```

Your program code must implement this divide-and-conquer algorithm as a *recursive* function named “SortAndCount”; code of the combine function, named “MergeAndCount”, must be in a separate function named as such. Your program should prompt for an input file name, and then read from the file a list of array elements one line at a time and display on the screen the number of inversions in the array and the sorted array elements. See sample_input.txt and sample_output.txt for examples of input and display output. Submit your source codes via Blackboard. Program codes should be working, and also neatly organized and well commented.

Last modified: March 28, 2021