Lab 1

1. Write code for insertion sort. (15%)

2. Write code for merge sort. (15%)

3. The running time of merge sort can be improved in practice by taking advantage of the fast running time of insertion sort when its input is "nearly" sorted. When merge sort is called on a subarray with no more than k elements, where k is a parameter, use insertion sort to sort the subarray. Write code for improved version of sorting algorithm which combines merge sort with insertion sort. (20%)

4. The improved version of sorting algorithm runs in $O(f(n, k))$ expected time, where n is the input size and k is the parameter defined in question 3. Show what is $f(n, k)$ by giving deduction. (20%)

5. Design experiments and write documentation for the questions above. The numbers to be sorted should be generated randomly, and you should consider different input sizes and different values of parameter k. (20%)

6. Let $A[1..n]$ be an array of n distinct numbers. If $i < j$ and $A[i] > A[j]$, then the pair (i, j) is called an *inversion* of A, where i and j are indices.
For instance, given an array {8, 1, 7, 10, 5, 9}, there are six inversions:
(1, 2), (1, 3), (1, 5), (3, 5), (4, 5), (4, 6).
Design an algorithm and write code for the algorithm that determines the number of inversions in any permutation on n elements in $\Theta(n \lg n)$ worst-case time. (Hint: Modify merge sort.) (10%)

You should hand in all the code for this lab and documents of the experiments and the answers to the questions above to ftp://10.132.141.33/classes/14/151 数据结构与算法设计(郑骁庆)/WORK_UPLOAD/lab1. All the files to be handed in should be packed into a zip or rar file with your student ID and name as the file name (e.g., 14302010001-冯泽灵.rar).

Deadline: 23:59 September 15th, 2015.
Late submission: 20% penalty per day. No marks for submitting 5 days after deadline.
Caution: No plagiarism is permitted. If spotted, both plagiarist and the one being plagiarized will get 0 marks for this lab and lose 5 marks in the total score of this course. We recommend those who would like to plagiarize not to submit their code.