

## ESO207: Assignment 3

Due on 27 August, 2015

**Note** Every time you give an algorithm pseudo-code, please ensure that you give sufficient explanation so one can understand easily. Also give the proof of correctness (show that it indeed computes the desired result) and provide the time complexity analysis (not just the time complexity).

**Q1.** Let  $\sigma = a_1 a_2 \dots a_n$  be a permutation of  $1 2 3 \dots n$ . Then the number of inversions of  $\sigma$  is the number of pairs  $(a_p, a_q)$  such that  $p < q$  but  $a_p > a_q$ .

Design a *divide and conquer* based algorithm which computes the number of inversions of a given permutation.

**Q2.** Given an array  $A$  of  $n$  integers stored in locations 0 to  $n - 1$  and a positive integer  $k$  less than  $n$ . Design an efficient algorithm using *divide and conquer* technique to compute the  $k$  largest integers from the array.

Note: Observe that you can sort the array and output the top  $k$  elements. That will require  $O(n \log n)$  time. So try to devise an algorithm which has better time complexity than  $O(n \log n)$ .