CSC 225 SPRING 2018
ALGORITHMS AND DATA STRUCTURES I
ASSIGNMENT 2
UNIVERSITY OF VICTORIA

1. Suppose that we are given an array A with n keys and k inversions. Here, an *inversion* is defined as a pair of entries that are out of order in the array. What is the running time of Insertion sort when it is used to sort A in Big Oh notation? Why?

2. Develop a $O(n \log n)$ algorithm for computing the number of inversions in a given array.

3. A permutation (or ranking) is an array of N integers where each of the integers between 0 and N-1 appears exactly once. The *Kendall tau distance* between two rankings is the number of pairs that are in different order in the two rankings. For example, the Kendall tau distance between 0 3 1 6 2 5 4 and 1 0 3 6 4 2 5 9 is four since the pairs 0-1, 3-1, 2-4 and 5-4 are in different relative order in the two rankings but all the other pairs are in the same relative order. Develop a $O(n \log n)$ algorithm for computing the Kendall tau distance between two rankings. (Hint: use the solution to the previous problem.)

4. Suppose we are given a sequence S of n elements, each of which is an integer in the range $[0; n^2 - 1]$. Describe a simple method for sorting S in $O(n)$ time.

5. Prove that it is impossible to develop a comparison-based implementation of the MinPQ ADT such that both *insert* and *delete the minimum* take only $O(\log \log n)$ time.