Homework 1

You will submit answers to a set of problems. All the answers can be in one file. You can

1. Write down the answers, as neat as possible, and scan your hand script.
2. Or use Word and its equations to type the answer (or any other text/math editor).
3. Or use LaTeX to create the file.

You should submit your homework before the deadline.

These are the problems for Homework 1: [10 points, each 2 points]

1- Explain ‘K-Dimension Tree’ data structure and discuss its strengths and limitations.

2- Suppose we are comparing implementations of insertion sort and merge sort on the same machine. For inputs of size n, insertion sort runs in $8n^2$ steps, while merge sort runs in 64n lg n steps. For which values of n does insertion sort beat merge sort?

3- Consider sorting n numbers stored in array A by first finding the smallest element of A and exchanging it with the element in A[1]. Then find the second smallest element of A, and exchange it with A[2]. Continue in this manner for the first n-1 elements of A. Write pseudocode for this algorithm, which is known as **selection sort**. What loop invariant does this algorithm maintain? Why does it need to run for only the first n - 1 elements, rather than for all n elements? Give the best-case and worst-case running times of selection sort in Θ notation.

4- **Inversion :** 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.

a. List the five inversions of the array {3,4,9,7,2}.

- **b.** What array with elements from the set {1,2,…,n} has the most inversions? How many does it have?
- **c.** What is the relationship between the running time of insertion sort and the number of inversions in the input array? Justify your answer.

5- **Ordering by asymptotic growth rates**

- **a.** Rank the following functions by order of growth; that is, find an arrangement $g_1, g_2, \ldots, g_{30}$ of the functions satisfying $g_1 = \Omega(g_2)$, $g_2 = \Omega(g_3)$, ..., $g_{29} = \Omega(g_{30})$. Partition your list into equivalence classes such that functions f(n) and g(n) are in the same class if and only if $f(n) = \Omega(g(n))$.

  Plain description : put the following functions in the descending order of the asymptotic growth rate.

n	$\ln \ln n$	2^n	$n \cdot 2^n$	$(n+1)!$	$\ln n$
1	e^n	$n!$	$2^{\lg n}$	$3^{2^{n+1}}$	n^3
$8^{\lg n}$	$(6/5)^n$	3^{2^n}	$n \lg n$	$n^{\lg \lg n}$	$(5/4)^n$

- **b.** Give an example of a single nonnegative function f(n) such that for all functions $g_i(n)$ in part (a), f(n) is neither $O(g_i(n))$ nor $\Omega(g_i(n))$.