CSE 2012- Design and Analysis of Algorithms
In-lab Practice Sheet 1

Practice makes you Perfect

1. Inputs required from the theory:
   (a) Pseudocode of Insertion sort

   (b) $T(n)$ of the insertion-sort

   (c) Best-case, worst-case and avaerage-case time-complexity of the algorithm

2. Design related problems
   (a) Translate the Insertion-sort algorithm discussed in the class into a program which takes n numbers (real or integers). Inputs for the program is n and the n numbers.

   (b) Given a sequence of n numbers (real or integers) and a number k (k is one among the n numbers), write an algorithm and the corresponding code to compute the position of k if the given n numbers are arranged in an increasing order, using insertion-sort. If the $2, -1, 3, 0, 7$ and 3 are the input , your program should output 4 since 3 will be in the fourth position (starting from 1), in the sorted (increasing) order. You are expected to code the problem two different ways, say, c_1 , c_2 using two different approaches. Decide whether c_1 is efficient or c_2 is efficient based on the running time $T(n)$ of the resepctive codes.

   (c) All the alphabets(lower case) of english language $a, b, c, ..., y, z$ are assigned values $1, 2, 3, ..., 25, 26$. Given a sequence of n symbols from english alphabet (only lower case), write an insertion-sort based algorithm to arrange the given n symbols, in an increasing order of their values. . You are expected to code the problem two different ways, say, c_1 , c_2 using two different approaches. Decide whether c_1 is efficient or c_2 is efficient based on the running time $T(n)$ of the resepctive codes.

(d) Given a sequence of n numbers (real or integers), write an algorithm and the corresponding code to arrange the given n numbers are arranged in such a way that all the negative numbers (if any) are arranged in a descending order and all the positive numbers are arranged in an increasing order with zero (if it is in the input) appearing between the smallest negative number and the smallest positive number. If $7, 3, 2, 4$ the output should be $2, 3, 4, 7$. If $-7, -3, 2, 4$ the output should be $-3, -7, 2, 4$ should be the output. If $7, 3, -1, 0, 2, 4$ the output should be $-1, 0, 3, 4, 7$.

(e) Given n points $P_1, P_2, ..., P_n$ with the coordinates $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$ respectively, write an insertion-sort based algorithm and the corresponding code to arrange the points in an increasing order of the distance of the point from the origin (with $(0, 0)$ as the coordinates. Distance between any two points (a_1, b_1) and (a_2, b_2) is $\sqrt{(a_1 - a_2)^2 + (b_1 - b_2)^2}$. Input for the code is : n and the coordinates of the n points entered with $x-$coordinate first and then the $y-$coordinate.

3. Analysis related problems

(a) We know that the running time of insertion-sort algorithm $T(n)$ is

$$
\begin{aligned}
T(n) &= c_1 n + c_2(n-1) + c_4(n-1) + c_5 \sum_{j=2}^{n} t_j + c_6 \sum_{j=2}^{n} (t_j - 1) \\
&\quad + c_7 \sum_{j=2}^{n} (t_j - 1) + c_8 (n-1) .
\end{aligned}
$$

where n is the size of the problem (number of inputs given), c_i is the time taken to execute the i-th line in the algorithm and t_j is the number of times the 'while-loop' is executed for the given input. Take the value of c_i's as 1. Execute the insertion-sort algorithm (through the respective code) and compute

$T(n)$ for different input size n. Based on the the values of n and $T(n)$, draw the graph with n as the x-axis and $T(n)$ as the y-axis. Based on the graph, conclude whether $T(n)$ is a linear or a parabola or a combination of line and a parabola.

(b) Execute the insertion-sort algorithm many time with different inputs and Justify the statement : Best-case running time of $T(n)$ is a linear function of n.

(c) Execute the insertion-sort algorithm many time with different inputs and Justify the statement : Worst-case running time of $T(n)$ is a quadratic function of n.

(d) Execute the insertion-sort algorithm many time with different inputs and Justify the statement : Average -case running time of $T(n)$ is a quadratic function of n.

(e) Compute the running time of P, t(P) in seconds for the insertion-sort algorithm for different inputs and draw the graph $n - Vs - \ t(p)$.

(f) Compute the $t(P)$ for all the codes requested in Q.No 2.