DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY

Algorithm Laboratory (CSLR41)

Assignment 2

Problem Statement: Implement Bubble sort, straight insertion sort and straight selection sort.

Bubble Sort Steps:

Step 1: $BOUND \leftarrow n$.
Step 2: $t \leftarrow 0$. *Perform Step 3 for* $i = 1, 2, \cdots, BOUND - 1$.
Step 3: *If* $K_i > K_{i+1}$ *then* $R_i \leftrightarrow R_{i+1}$ *and* $t \leftarrow i$.
Step 4: *If* $t = 0$, *terminate the algorithm.*
Otherwise $BOUND \leftarrow t$ *and go to Step 2.*

Straight Insertion Sort Steps:

Step 1: *Repeat Step 2 to Step 5 for* $j = 2, 3, \cdots, n$.
Step 2: $i \leftarrow j - 1, K \leftarrow K_j, R \leftarrow R_j$.
Step 3: *If* $K \geq K_j$ *go to Step 5.*
Step 4: $K_i \leftarrow K_{i-1}, R_i \leftarrow R_{i-1}, i \leftarrow i - 1$. *If* $i > 0$, *then go to Step 3.*
Step 5: $K_i \leftarrow K, R_i \leftarrow R$.

Straight Selection Sort Steps:

```
1. Set MIN to location 0.
2. Search the minimum element in the list.
3. Swap with value at location MIN.
4. Increment MIN to point to next element.
5. Repeat until the list is sorted.
```

Input: *n* random integers, K is another random integer where n = 10, 100, 1000, 10000, 100000.

Output: For each of these implementations do the following:

1. Analyze the behavior for best case, worst case and some random cases.
2. Plot and find the time complexity in terms of asymptotic notion for all these three cases by varying input size and noting down the time required for sorting.
3. Take f(n)=$c_1 n^2$ and g(n) =$c_2 n^2$. Plot the graph for the worst case of the algorithm along with these two functions. Find some constants for f(n) and g(n) such that the plot is bounded by f(n) and g(n) by above and below respectively.
4. Compare performance of your sorting algorithms with the in-build sort function.
5. Write your observations and derive possible conclusions.