OS- Assignment

Anupreet Singh
11911063
CSE

Q-1- Insertion Sort

Algorithm :-

```
Insertion Sort (int arr[], size)
{
    for(i -> size
     j = arr[i];
    K = j - 1
    while (K >= 0 && arr[K] > j)
    {arr[K+1] = arr[K]
      K--;
    }
    arr[K+1] = j
}
```


Since, Insertion Sort is Modifying the original array by inserting the lower element at the right place in the original array only. Thus, it does not require any extra space. Hence, it is an "In-Place Sorting" Algorithm.

∴ Space complexity = O(1).

2 Basic operation takes place in the algorithm :-
i) Comparison ii) Swapping

In Best case i.e. the array is already sorted. This algorithm only compares 'n' elements.

∴ Time complexity = O(n)

Ans-2- Quick sort :-

Divide and Conquer Algorithm.

Time Complexity

1) Worst Case :-

$T(n) = O(n^2)$

2.) Avg. Case

$T(n) = O(n \log n)$

3). Best Case

$T(n) = O(n \log n)$

→ Bubble Sort

Time Complexity

For n elements, $(n-1)$ comparisons are done

$T(n) = 1+2+3 \ldots (n-1)$

$T(n) = \frac{n(n-1)}{2} = \frac{n^2-n}{2}$

$O(n^2)$

→ Both, Quick Sort and Bubble Sort algorithm are "In-Place" Algorithm

→ Bubble sort is efficient for small size arrays.

Time complexity for Merge Sort → $O(n \log n)$

Time complexity for Insertion Sort → $O(n^2)$