Name - Abhinav Bhatt
Branch - CSE
Roll no - 11911055

Explantions

Qs. Perform Analyses on time complexity - .... lower order term.

Sol.

Case I

Best case Time complexity of Insertion Sort.

It happens only when the elements given are itself in sorted order. as by the program we can see there are two for loops

```
for ( i=1;  i<n ; i++)
  {
 k = arr[i]
   for (j= i-1;   j>=0 && K< arr[j];
                              j--)
```

Best Ex-

Elements- 0 1 2 3 4.

$K = arr[1] = 1.$

& $arr[j] = arr[0] = 0.$

So we can easily see that for ~~whole~~ all $i < 5$, inner most for loop will not be executed
Or we can say that the execution is only 1 time.

Time complexity $= O(n-1)$
$\approx O(n)$

Q2-

Giving suitable example . . . . .
. . . Also make program of each algorithms.

Quick Sort (Worst case)

10 20 30 40 50

find element greter than 10

Find element smaller than 10

Teacher's Sign. ________

main $\boxed{10}$ 20 30 40 50

j i

swap the main element with j th element.

1 + n

swap comparisions

=> 20 30 40 50

do step 1 with 20 do step 2 with 20

20 30 40 50 ∞

j i

do step 3

$\underline{1}$ + (n-1)

swap. comparisions

. 30 40 50

do step 1 with 30 do step 2 with 30

30 40 50

j i

Teacher's Sign.

Scanned by CamScanner

do step 3   1 + n-2

swap comparisions

= 40   50

do step 1 with 10   do step 2 with 40.

40   50

1   i

Abhinav

1+   n-3

swap   comparisions

50

If a list contain n element then time complexity

$= 1 + n + 1 + n-1 + 1 + n-2 + \cdots 1 + 1.$

$= (1 + 1 + \cdots n \text{ times}) + (n + n + \cdots 1)$

$= n + \frac{n(n+1)}{2}$

$= \frac{2n + n^2 + n}{2}$

Teacher's Sign. ________

[illegible]

$= \frac{[illegible]}{n}$

$= O(n^2)$ (worst case)

As it does not need an extra space so it inplace algorithm.

Algo name	Cases	Time
Quick sort	Best	$O(n \log n)$
	Avg	$O(n^2 \log n)$
	Worst	~~$O(n \log n)$~~ $O(n^2)$
Merge sort	Best	$\left.\begin{matrix} \\ \\ \\ \end{matrix}\right\}$ $O(n \log n)$
	Avg	
	Worst	
insertion sort	Best	$O(n)$
	Avg	$O(n^2)$
	worst	$O(n^2)$
Bubble sort	Best	$O(n)$
	Avg	$O(n^2)$
	Worst	$O(n^2)$

Bubble sort

```
Void Bubble sort ( int arr[], int n)
  {
    int i, j
    for ( i=0; i <n-1 ; i++)
      {
        for ( j=0 ; j<n-i-1; j++ )
          {
            if (arr [j] > arr [ j+1])
            swap ( arr[j] , &arr [j+1])
          }
      }
```

Worst case

when	inner loop run
~~i=0~~	
~~i=1~~	
j=0	n times
i=1	n-1 "
i=2	n-2 "
.	
:	
i=n-2	2 times

Teacher's Sign. ____

Total = n-1 + n-2 + n-3 + ... 2+1-1

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

$= O(n^2)$

Best case = O(n) (when element are already sorted)

Avg case = O(n²)

Abhinav