Name → SOMBIR

Roll no. → 11911025

Branch → C.S.E.

Best Case time complexity
of Insertion sort

Given array : 2, 3, 4, 5

i	0	1	2	3	4
Element					
j					
key			3	4	5
no. of Comparision			0	0	1
Movements			1	1	1

Time complexity $= 1 + 1 + 1 = 3$

$= O(n-1)$

$\approx O(n)$

② find elements smaller than 10.

main ⑩   20   30   40   50.

      j   i

Step :③ swap the main element

     with j^{th} element.

       1 +   n

      <u>swap</u>   comparison

⟹ 20   30   40   50   🗸

do step 1   do step 2

with 20.   with 20

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
$i = 0$	n times
$i = 1$	$n-1$ times
$i = 2$	$n-2$ times
$\vdots$	
$i = n-2$	2 times

Total $= n + n-1 + n-2 + n-3 + \dots + 2$

$= \dfrac{n(n+1)}{2} - 1 \approx O(n^2)$

40   50

do step 1   do step 2
with 40   with 40

                                        (Samir)

40   50

j   i

1   +   n-3
$\underbrace{swap}$ $\underbrace{comparisons}$

If a list contain 'n' elements then
time complexity:-

$= (1+n) + (1+n-1) + 1+n-2 +1+n-3+ --- +1+$

$= (\underbrace{1+1+ --- +1}_{n\ times}) + (n+n-1+n-2+ --- +1)$

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

$= \dfrac{2n+n^2+n}{2} = \dfrac{3n+n^2}{2}$

$\simeq O(n^2)$ worst case

Algorithm name	Cases	Time
Quick Sort	Best	$n \log n$
	Average	
	Worst	n^2
Merge Sort	Best case	$n \log n$
	Average	
	worst	

Algorithm name	Cases	Time
Quick sort	Best	
	Average	$n \log n$
	Worst	n^2
Merge sort	Best case	
	Average	$n \log n$
	worst	
Insertion sort	Best	n
	Average	n^2
	Worst	n^2
Bubble sort	Best	n
	Average	n^2
	Worst	n^2

Sombir