

Name: GOVIND KUMAR

Branch: IT

Roll NO: 11912057

Assignment (sorting and BST).

① solⁿ: Analysis on the time complexity of insertion sort algorithm in best case.

Let A be the array of size '4' and its item in sorted order are 2, 3, 4, 5. we taken elements in sorted order because this is the best case scenario.

$$A[] = \{2, 3, 4, 5\};$$

	0	1	2	3	4
i		1	2	3	
Element		2	3	4	5
J			2	3	4
Key			3	4	5
Comparison			1	1	1
movement			0	0	0

$$\text{time complexity} = 1 + 1 + 1 + \dots + (n-1)$$

$$= O(n)$$

We can use binary search to reduce the number of comparisons in normal insertion sort. Binary insertion sort uses binary search to find the proper location to insert the selected item at each iteration. In normal insertion sort, it takes $O(n^2)$ comparisons (at nth iteration) in worst case. we can reduce it to $O(n \log n)$ by using binary search.

And also use doubly linked list, linear search to reduce time complexity of insertion sort from $O(n^2)$ to some lower order term.