

21BCE7371

RADHA KRISHNA GARG

DSA DIGITAL ASSIGNMENT-1

Ans-1

Time Complexity of the First 'for' Loop: $O(n)$

Time Complexity of the Second (Nested) 'for' Loop: $O(\log n)$

Therefore, Time Complexity of the Function: $n \times \log n$

$O(n \log n)$

Ans-2

The provided sorting algorithm is known as *Insertion Sort* and it's

Time Complexity is: $O(n^2)$

Ans-3

A simple *Bubble Sort* will be sufficient. Time Complexity: $O(n^2)$

Ans-4

Function to add element in the middle of the singly linked list.

```
static void addMid(int d){ Node temp = new Node(d); if (head == null)
    head = temp; else {
    Node ptr = head; int len = 0;
    while (ptr != null) { len++;
    ptr = ptr.next;
    int c=((len%2)==0)?(len/2):(len+1)/2;    //c to store index to add element at
    ptr = head; while (c-- > 1)
    ptr = ptr.next;
    temp.next = ptr.next;    //updating the list at stored index ptr.next = temp;
    }
}
```

Ans-5

Source code

```
public void sortList(){ //node current will point to head Node current = head, index = null;
    int temp;
    if (head == null) { return;
    }
    else {
        while (current != null) { //node index will point to node next to
            current
            index = current.next;
            while (index != null) { //if current node's data is greater than index's node data, swap the data between them
                if (current.data > index.data) { temp = current.data; current.data = index.data; index.data = temp;
                }
                index = index.next;
            }
            current = current.next;
        }
    }
}
```

Time Complexity: $O(n^2)$ [bubble sort]

Ans-6

Assumption – list is sorted.

```
public void insSort(int d){
    Node temp=new Node(d); //node to store the new node Node cur=head; //node to traverse to required node
    while(cur.next.data<d || cur.next!=null){ //loop to traverse to required node
        cur=cur.next;
    }
    temp.next=cur.next; //updating the list cur.next=temp;
}
```

Time Complexity: $O(n)$ {n being the iterations of the while loop}