

**DSA-Lab Tasks**

**Task\_01**

**Explanation:**

* **n is a normal integer.**
* **myPtr store the memory address of n.**
* **using \*myPtr, we can modify n without directly accessing it.**
* **The output will show the n value and modified value also.**

**Task\_02**

**Explanation:**

* **First we create a function findMaximum.**
* **We initialize an array with 0 index.**
* **Inside the function the for loop runs to check the highest value and returns maximumValue.**
* **Inside the main function we declare the values of array and then use building function sizeof for checking the size of array and performing a function. At last we print the maximum value.**
* **The time complexity of this function is O(n). Because loop inside the function runs through n element.**

**Task\_03**

**Explanation:**

**Singly linked list is a sequence of nodes where each node contains data and pointer to the next node.**

**In this task we implement two insertion methods:**

**InsertAtStart: insert a new node at start becomes first node.**

**InsertAtLast: new node is added at last.**

* **Inserting at start, updates the new node's next pointer to point to the od head.**
* **Inserting at last traverse to the last node and update next pointer.**
* **Updated lists are displayed in display function.**

**Task\_04**

**Explanation:**

**In this task we need to insert a node at a given position in our linked list.**

* **Inserting at start at position 1 treat it as inserting at start.**
* **Traverse the position before the requirement index.**
* **Update the pointer to insert new node at the correct location.**
* **Handling case position where is out of range.**
* **And last again display the modified nodes in display function.**

**Task\_05**

**Explanation:**

* **Function we need to display the nodes.**
* **First node head of the list, last node traverse to the last, nth node traverse to the given position and last center node.in this we use slow and fast Pointers method.**
* **First node points to the head's data, last node traverse to the last and it data, nth traverse to it and it data and points to it, as so center node.**