```c
//insert data at the end of the linked list. section j,p -maruf
#include<stdio.h>
#include<stdlib.h>

struct Node
{
    int data;
    struct Node* next;
};
struct Node* head; //Globally we have declared the head so that we can access to head from anywhere of the
//program

void insert_number(int x)
{
    struct Node* temp = (struct Node*)malloc(sizeof(struct Node)); //dynamically allocate the memory
    temp->data = x;
    temp->next = NULL;
    struct Node* temp1 = head; // New Node pointer is created to store the previous Node Addres .
    if(head == NULL){ // previously if list is empty
        head = temp; // when list is empty and a new data is added then head contains the new node address
    }
    else{
        while(temp1->next!=NULL){ // traverse untill the last Node
            temp1 =temp1->next;
        }
        //temp1 = Now you are at the last Node of the old list
        temp1->next = temp; // insert the new Node address to the previous last Node->link

    }

}

void display()
{
    struct Node* temp = head;
    printf("List contains: ");
    while(temp != NULL)
    {
        printf("%d ", temp->data);
        temp = temp->next; // pointing the next Node
    }
    printf("\n");
}

int main()
{
    head = NULL;
    int number,i, x;
    printf("Enter the size of the list\n");
    scanf("%d", &number);
    for(i=0; i<number; i++ )
    {
        printf("Enter Number to Insert\n");
        scanf("%d",&x);
//to insert number at beginning
        insert_number(x);
//To print the whole list. where the last entered data will be in first position of the list
        display();

    }
    return 0;
}
```