```c
#include<stdio.h>
#include<stdlib.h>
struct node
{
    int data;
    struct node *next;
};
struct node *start=0;
struct node *createNode()
{
    struct node *cn;
    cn = (struct node*)malloc(sizeof(struct node));
    printf("\nEnter data:");
    scanf("%d",&cn->data);
    cn->next = 0;
    return cn;
}
void insertStart()
{
    struct node *is;
    is = createNode();
    if(start == 0)
    {
        start = is;
    }
    else
    {
        is->next = start;
        start=is;
    }
}
void insertMiddle()
{   int data;
    struct node *t2, *k3;
    t2=createNode();
    printf("Enter the data after which you want to insert: ");
    scanf("%d", &data);
    if(start==0)
    {
        start=t2;
    }
    else
    {
        k3=start;
        while(k3->data!=data)
        {
            k3=k3->next;
        }
        t2->next=k3->next;
        k3->next=t2;

    }

}
void insertEnd()
{
    struct node *ie,*search;
    ie=createNode();
    if(start == 0)
    {
        start = ie;
    }
    else
    {
        search=start;
        while(search->next != 0)
        {
            search=search->next;
        }
        search->next = ie;
    }
}
void deleteStart()
{
    struct node *ds;
    if(start == 0)
    {
        printf("\nNo Data found for delete.");
    }
    else
    {
        ds = start;
        start = start->next;
        ds->next=0;
        free(ds);
        printf("\nDeleted Succesfully");
    }
}
void deleteMiddle()
{
    struct node *dm,*search;
    int loc_data;
    if(start==0 || start->next == 0)
    {
        printf("\nNot Sufficient data found to apply this operation.");
        return;
    }
    search=start;
    printf("\nEnter the data that you want to delete :");
    scanf("%d",&loc_data);
    if(start->data == loc_data)
    {
        printf("\nPlease apply delete from start method.");
        return;
    }
    while(search->next->data != loc_data)
    {
        search=search->next;
    }
    dm = search->next;
    search->next=dm->next;
    dm->next=0;
    free(dm);
    printf("\nDeleted Succesfully");
}
void deleteEnd()
{
    struct node *search,*de;
    if(start == 0)
    {
        printf("\nNo Data found for delete.");
        return;
    }
    if(start->next == 0)
    {
        de=start;
        start=0;
        free(de);
    }
    else
    {
        search=start;
        while(search->next->next != 0)
        {
            search=search->next;
        }
        de=search->next;
        search->next=0;
        free(de);
    }
    printf("\nDeleted Succesfully.");
}
void display()
{
    struct node *ds;
    if(start == 0)
    {
        printf("\nNo data found in list.");
    }
    else
    {
        ds=start;
        printf("\nYour data = [");
        while(ds != 0)
        {
            printf("%d,",ds->data);
            ds = ds->next;
        }
        printf("]");
    }
}
int main()
{
    int ch;
    printf("1. Insert at Start.\n");
    printf("2. Insert at middle.\n");
    printf("3. Insert at end.\n");
    printf("4. Delete from Start.\n");
    printf("5. Delete from Middle.\n");
    printf("6. Delete from End.\n");
    printf("7. Display.\n");
    printf("8. Exit.");
    while(1)
    {
        printf("\n\n>>Enter your choice :");
        scanf("%d",&ch);
        switch(ch)
        {
            case 1:
                insertStart();
                break;
            case 2:
                insertMiddle();
                break;
            case 3:
                insertEnd();
                break;
            case 4:
                deleteStart();
                break;
            case 5:
                deleteMiddle();
                break;
            case 6:
                deleteEnd();
                break;
            case 7:
                display();
                break;
            case 8:
                exit(0);
                break;
            default:
                printf("You have entered a invaid input.");
        }
    }
    return 0;
}
```