```c
//Linked List 6
#include <stdio.h>
#include <stdlib.h>

struct node{
    int data;
    struct node *next;
};

void insertlast(struct node **headptr){
    struct node *newnode,*temp;
    int value;
    printf("Enter value: ");
    scanf("%d",&value);
    newnode = (struct node*)malloc(sizeof(struct node));
    newnode->data = value;
    newnode->next = NULL;
    temp = (*headptr);
    if(*headptr == NULL)
        (*headptr) = newnode;
    else{
        while(temp->next != NULL)
            temp = temp->next;
        temp->next = newnode;
    }
}
void deletefirst(struct node **headptr){
    if((*headptr) == NULL)
        printf("The list is empty\n");
    else if((*headptr)->next == NULL)
        (*headptr) = NULL;
    else{
        (*headptr) = (*headptr)->next;
    }
}
void deletepos(struct node **headptr){
    struct node *temp;
    int count=0,currpos=1,pos;
    if((*headptr) == NULL){
        printf("The list is empty\n");
        return;
    }
    else{
        if((*headptr)->next == NULL){
```

```
            (*headptr) = NULL;
            printf("The only element in the list was deleted\n");
            return;
        }
        temp = (*headptr);
        while(temp != NULL){
            count++;
            temp = temp->next;
        }
        printf("There are %d elements in the list.Enter pos of element to be deleted: ",count);
        scanf("%d",&pos);
        if(pos > (count+1)){
            printf("No such position is present\n");
            return;
        }
        if(pos == (count+1)){
            temp = (*headptr);
            while((temp->next)->next != NULL)
                temp = temp->next;
            temp->next = NULL;
        }
        else{
            temp = (*headptr);
            while(temp->next != NULL){
                if(pos == 1){
                    (*headptr) = (*headptr)->next;
                    return;
                }
                if(currpos == pos-1){
                    temp->next = (temp->next)->next;
                    return;
                }
                currpos++;
                temp = temp->next;
            }
            printf("No such element was found!!!\n");
        }
    }
}
void deletelast(struct node **headptr){
    struct node *temp;
    temp = (*headptr);
    if((*headptr) == NULL)
        printf("The list is empty\n");
```

```
    else if((*headptr)->next == NULL)
        (*headptr) = NULL;
    else{
        temp = *headptr;
        while((temp->next)->next != NULL)
            temp = temp->next;
        temp->next = NULL;
    }
}
void display(struct node *temp){
    if(temp == NULL){
        printf("The list is empty\n");
        return;
    }
    else{
        while(temp != NULL){
            printf("%d\t",temp->data);
            temp = temp->next;
        }
        printf("\n");
    }
}

int main(int argc,char **argv){
    int choice;
    struct node *head = NULL;
    while(choice != 6){
        printf("Enter choice 1)insertlast 2)deletefirst 3)deletepos 4)deletelast 5)display 6)exit : ");
        scanf("%d",&choice);
        switch(choice){
            case 1:insertlast(&head);break;
            case 2:deletefirst(&head);break;
            case 3:deletepos(&head);break;
            case 4:deletelast(&head);break;
            case 5:display(head);break;
            case 6:
            default:exit(0);
        }
    }
    return 0;
}
```