```c
//Linked List 5
#include <stdio.h>
#include <stdlib.h>

struct node{
    int data;
    struct node *next;
};
void insertfirst(struct node **headptr){
    struct node *newnode;
    int value;
    printf("Enter value: ");
    scanf("%d",&value);
    newnode = (struct node*)malloc(sizeof(struct node));
    newnode->data = value;
    newnode->next = NULL;
    if(*headptr == NULL)
        (*headptr) = newnode;
    else{
        newnode->next = (*headptr);
        (*headptr) = newnode;
    }
}
void insertpos(struct node **headptr){
    struct node *newnode,*temp;
    int count=0,currpos=1,value,pos;
    printf("Enter value: ");
    scanf("%d",&value);
    newnode = (struct node*)malloc(sizeof(struct node));
    newnode->data = value;
    newnode->next = NULL;
    if((*headptr) == NULL){
        (*headptr) = newnode;
    }
    else{
        temp = (*headptr);
        while(temp != NULL){
            count++;
            temp = temp->next;
        }
        printf("There are %d elements in the list.Enter the position where you want to insert the value: ",count);
        scanf("%d",&pos);
        if(pos > (count+1)){
            printf("no such position\n");
```

```
            return;
        }
        if(pos == count+1){
            temp = (*headptr);
            while(temp->next != NULL)
                temp = temp->next;
            temp->next = newnode;
        }
        else{
            temp = (*headptr);
            while(temp->next != NULL){
                if(currpos == (pos-1)){
                    newnode->next = temp->next;
                    temp->next = newnode;
                    break;
                }
                currpos++;
                temp = temp->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 deletelast(struct node **headptr){
    struct node *temp;
    temp = (*headptr);
    if((*headptr) == NULL)
        printf("The list is empty\n");
```

```c
    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)insertfirst 2)insertpos 3)insertlast 4)deletelast 5)display 6)exit : ");
        scanf("%d",&choice);
        switch(choice){
            case 1:insertfirst(&head);break;
            case 2:insertpos(&head);break;
            case 3:insertlast(&head);break;
            case 4:deletelast(&head);break;
            case 5:display(head);break;
            case 6:
            default:exit(0);
        }
    }
    return 0;
}

```