```
#include <stdio.h>
#include <stdlib.h>
void create();
void display();
void delfun(int);
void front_delete(int);
void end_delete();
struct node
{
int data;
struct node *next;
};
struct node *head=NULL;
int c=0;
int main(int argc, char **argv)
{
       int choice;
       int ele;
       do
      {
     printf("\n1. Create \n2. Display \n3. Delete specified element \n4. Delete at
beginning \n5. Delete at End");
     printf("\nEnter your choice : ");
     scanf("%d",&choice);
     switch(choice)
      {
         case 1: create(); break;

         case 2: display();break;

         case 3: printf("Enter the element you want to delete::\n");
         scanf("%d",&ele);
         delfun(ele);
         break;

         case 4: front_delete(1);
         break;

         case 5:
         end_delete();
         break;
        default:exit(0);
      }
    } while(choice<=5);
}
void create()
{
    struct node *newnode,*temp;
```

```
    int item;
    newnode =(struct node *) malloc (sizeof(struct node));
    printf("Enter the data : ");
    scanf("%d",&item);
     newnode->data=item;
    if (head==NULL)
    {
       newnode->next=NULL;
       head=newnode;
       printf("Node created\n");
    }
    else
    {
       temp=head;
       while(temp->next!=NULL)
        {
        temp=temp->next;
        }
        temp->next=newnode; newnode->next=NULL;
        printf("Node created\n");
    }
}
void display()
{
      struct node *ptr=NULL;
      ptr=head;
      if(ptr==NULL)
     {
          printf("Nothing to print\n");
      }
    else
    {
       while(ptr!=NULL)
       {
        printf("%d ",ptr->data);
        ptr=ptr->next;
       }
    }
}
void delfun(int ele)
{
    struct node *temp,*prev=NULL;
     if (head == NULL)
     {
         printf("Empty List. Can't delete\n");
         return;
     }
     temp=head;
```

```
     if(head->data==ele)
     {
         head=head->next;
         return;
     }
    while (temp != NULL && temp->data != ele)
    {
        prev = temp;
        temp = temp->next;
    }
  // If key was not present in linked list
    if (temp == NULL)
    {
        printf("Element not found in the list\n");
        return;
    }
// Unlink the node from linked list
    prev->next = temp->next;
    free(temp);
    return;
}
void front_delete(int n)
{
    if (head == NULL)
     {
         printf("Empty List. Can't delete\n");
         return;
     }
    struct node* temp1=head;
    if(n==1)
    {
        head=temp1->next;
        free(temp1);
        printf("Front node deleted\n");
        return;
    }
}
void end_delete()
    {
        struct node *ptr,*ptr1;
        if(head == NULL)
        {
            printf("list is empty\n");
        }
        else if(head -> next == NULL)
        {
            head = NULL;
            free(head);
```

```
    printf("Only node of the list deleted \n");
}
else
{
    ptr = head;
    while(ptr->next != NULL)
        {
            ptr1 = ptr;
            ptr = ptr ->next;
        }
        ptr1->next = NULL;
        free(ptr);
        printf("Deleted last Node\n ...");
    }
}
```