```
//Double Linked List
#include<stdio.h>
#include<stdlib.h>
struct node
{
 int data;
 struct node*prev,*next;
}*h;
void addbeg()
{
 int item;
 struct node*new;
 new=(struct node *) malloc (sizeof(struct node *));
 printf("Enter the data: ");
 scanf("%d",&item);
 new->data=item;
 new->next=NULL;
 new->prev=NULL;
 if(h->next==NULL)
 {
  h->next=new;
  new->prev=h;
 }
 else
 {
  new->next=h->next;
  new->next->prev=new;
  h->next=new;
  new->prev=h;
 }
}
void addbtw()
{
 int item,key;
 struct node *new,*ptr;
 new=(struct node *) malloc (sizeof(struct node *));
 printf("Enter the data: ");
 scanf("%d",&item);
 printf("\nEnter the key: ");
 scanf("%d",&key);
 new->data=item;
 new->next=NULL;
 new->prev=NULL;
 ptr=h->next;
 while(ptr->data!=key&&ptr->next!=NULL)
 {
  ptr=ptr->next;
 }
 if(ptr->next==NULL)
 {
  printf("Insertion not possible");
 }
 else
 {
  new->next=ptr->next;
  new->next->prev=new;
  ptr->next=new;
  new->prev=ptr;
 }
}
void addend()
{
 int item;
 struct node *new,*ptr;
 new=(struct node *)malloc(sizeof(struct node *));
 printf("\nEnter the data: ");
 scanf("%d",&item);
 new->data=item;
 new->next=NULL;
 new->prev=NULL;
 ptr=h->next;
 while(ptr->next!=NULL)
 {
  ptr=ptr->next;
 }
 ptr->next=new;
 new->prev=ptr;
}
void disp()
{
 struct node *ptr;
 ptr=h->next;
 if(ptr==NULL)
 {
  printf("List Empty\n");
 }
 else
 {
  while(ptr!=NULL)
  {
   printf("%d\t",ptr->data);
   ptr=ptr->next;
  }
 }
}
void delbeg()
{
 struct node *ptr;
 ptr=h->next;
 if(ptr==NULL)
 {
  printf("\nList Empty");
 }
 else
 {
  h->next=ptr->next;
  ptr->next->prev=h;
  free(ptr);
 }
}
void delend()
{
 struct node *ptr;
 ptr=h->next;
 if(ptr==NULL)
 {
  printf("\nList Empty");
 }
 else
 {
  while(ptr->next!=NULL)
  {
   ptr=ptr->next;
  }
  ptr->prev->next=NULL;
  free(ptr);
 }
}
void delbtw()
{
 int key;
 printf("\nEnter the key: ");
 scanf("%d",&key);
 struct node *ptr;
 ptr=h->next;
 if(ptr==NULL)
 {
  printf("\nList Empty");
 }
 else
 {
  while(ptr->data!=key&&ptr->next!=NULL)
  {
   ptr=ptr->next;
  }
  if(ptr->next==NULL)
  {
   printf("\nDeletion in between not possible");
  }
  else
  {
   ptr->prev->next=ptr->next;
   ptr->next->prev=ptr->prev;
   free(ptr);
  }
 }
}
void main()
{
 int ch;
 h=(struct node *)malloc(sizeof(struct node *));
 do
 {
      printf("\n1:Add beginning\n2:Add between\n3:Add ending\n4:Display\n5:Del beginning\n6:Del ending\n7:Del between\n8:EXIT\n");
  printf("Enter the choice: ");
  scanf("%d",&ch);
  switch(ch)
  {
   case 1:addbeg();break;
   case 2:addbtw();break;
   case 3:addend();break;
   case 4:disp();break;
   case 5:delbeg();break;
   case 6:delend();break;
   case 7:delbtw();break;
   case 8:exit(0);break;
   default:printf("\nInvalid choice");
  }
 }while(ch<=8);
}

/*OUTPUT:

1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
7:Del between
8:EXIT
Enter the choice: 4
List Empty

1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
7:Del between
8:EXIT
Enter the choice: 1
Enter the data: 10

1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
```

```
7:Del between
8:EXIT
Enter the choice: 4
10
1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
7:Del between
8:EXIT
Enter the choice: 1
Enter the data: 5

1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
7:Del between
8:EXIT
Enter the choice: 4
5 10
1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
7:Del between
8:EXIT
Enter the choice: 3

Enter the data: 20

1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
7:Del between
8:EXIT
Enter the choice: 4
5 10 20
1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
```

```
7:Del between
8:EXIT
Enter the choice: 2
Enter the data: 15

Enter the key: 10

1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
7:Del between
8:EXIT
Enter the choice: 4
5 10 15 20
1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
7:Del between
8:EXIT
Enter the choice: 3

Enter the data: 25

1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
7:Del between
8:EXIT
Enter the choice: 4
5 10 15 20 25
1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
7:Del between
8:EXIT
Enter the choice: 5

1:Add beginning
2:Add between
3:Add ending
4:Display
```

```
5:Del beginning
6:Del ending
7:Del between
8:EXIT
Enter the choice: 4
10 15 20 25
1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
7:Del between
8:EXIT
Enter the choice: 6

1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
7:Del between
8:EXIT
Enter the choice: 4
10 15 20
1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
7:Del between
8:EXIT
Enter the choice: 7

Enter the key: 15

1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
6:Del ending
7:Del between
8:EXIT
Enter the choice: 4
10 20
1:Add beginning
2:Add between
3:Add ending
4:Display
5:Del beginning
```

```
6:Del ending
7:Del between
8:EXIT
Enter the choice: 8
user@user-WIV68B55-0113:~$ */
```