```
#include<stdio.h>
#include<stdlib.h>
struct node
{
	int data;
	struct node *next,*pre;
};
struct node * create_dl(struct node *);
void print(struct node *);
struct node * insert_beg(struct node *);
struct node * insert_end(struct node *);
struct node * insert_val(struct node *);
struct node * insert_pos(struct node *);
struct node * delete_beg(struct node *);
struct node * delete_end(struct node *);
struct node * delete_val(struct node *);
struct node * delete_pos(struct node *);
void main()
{
	struct node *dl=NULL;
	int ch;
	printf("Enter the choice:\n");
	scanf("%d",&ch);
	switch(ch)
	{
		case 1 : dl=create_dl(dl);
```

```
        print(dl);
        break;
case 2 : dl=insert_beg(dl);
        print(dl);
        break;
case 3 : dl=insert_end(dl);
        print(dl);
        break;
case 4 : dl=insert_val(dl);
        print(dl);
        break;
case 5 : dl=insert_pos(dl);
        print(dl);
        break;
case 6 : dl=delete_beg(dl);
        print(dl);
        break;
case 7 : dl=delete_end(dl);
        print(dl);
        break;
case 8 : dl=delete_val(dl);
        print(dl);
        break;
case 9 : dl=delete_pos(dl);
        print(dl);
        break;
```

```
        }
}
struct node * create_dl(struct node *dl)
{
        struct node *p,*last;
        int i,n,ele;
        printf("Enter n value:\n");
        scanf("%d",&n);
        for(i=0;i<n;i++)
        {
                p=(struct node *)malloc(sizeof(struct node));
                printf("Enter the element:\n");
                scanf("%d",&ele);
                p->data=ele;
                p->pre=NULL;
                p->next=NULL;
                if(dl==NULL)
                {
                        dl=p;
                        last=p;
                }
                else
                {
                        p->pre=last;
                        last->next=p;
                        last=p;
```

```
            }
        }
        return dl;
}
void print(struct node *dl)
{
        while(dl!=NULL)
        {
            printf("Element=%d\n",dl->data);
            dl=dl->next;
        }
}
struct node * insert_beg(struct node *dl)
{
        dl=create_dl(dl);
        int element;
        struct node *p;
        printf("Enter an element:\n");
        scanf("%d",&element);
        p=(struct node *)malloc(sizeof(struct node));
        p->data=element;
        p->pre=NULL;
        p->next=NULL;
        p->next=dl;
        dl->pre=p;
        dl=p;
```

```
	return dl;
}
struct node * insert_end(struct node *dl)
{
	dl=create_dl(dl);
	struct node *p,*q;
	int element;
	printf("Enter an element:\n");
	scanf("%d",&element);
	p=(struct node *)malloc(sizeof(struct node));
	p->data=element;
	p->pre=NULL;
	p->next=NULL;
	q=dl;
	while(q->next!=NULL)
	{
		q=q->next;
	}
	q->next=p;
	p->pre=q;
	return dl;
}
struct node * insert_val(struct node *dl)
{
	dl=create_dl(dl);
	int val,element;
```

```
        struct node *p,*q;
        printf("Enter the value:\n");
        scanf("%d",&val);
        printf("Enter an Element:\n");
        scanf("%d",&element);
        p=(struct node *)malloc(sizeof(struct node));
        p->data=element;
        p->pre=NULL;
        p->next=NULL;
        q=dl;
        while(q!=NULL&&q->data!=val)
        {
              q=q->next;
        }
        p->next=q->next;
        p->pre=q;
        q->next->pre=p;
        q->next=p;
        return dl;
}
struct node * insert_pos(struct node *dl)
{
        dl=create_dl(dl);
        int pos,element,i;
        struct node *p,*q;
        p=(struct node *)malloc(sizeof(struct node));
```

```
printf("Enter position:\n");
scanf("%d",&pos);
printf("Enter an element:\n");
scanf("%d",&element);
p->data=element;
p->pre=NULL;
p->next=NULL;
q=dl;
if(pos==1)
{
	p->next=dl;
	dl->pre=p;
	dl=p;
}
else
{
	for(i=0;i<pos-2;i++)
	{
		q=q->next;
	}
	p->next=q->next;
	p->pre=q;
	q->next->pre=p;
	q->next=p;
}
return dl;
```

```
}
struct node * delete_beg(struct node *dl)
{
	dl=create_dl(dl);
	struct node *p;
	p=dl;
	dl->next->pre=NULL;
	dl=dl->next;
	free(p);
	return dl;
}
struct node * delete_end(struct node *dl)
{
	dl=create_dl(dl);
	struct node *p,*q;
	q=dl;
	while(q->next->next!=NULL)
	{
		q=q->next;
	}
	p=q->next;
	q->next->pre=NULL;
	q->next=NULL;
	free(p);
	return dl;
}
```

```
struct node * delete_val(struct node *dl)
{
	dl=create_dl(dl);
	int val;
	struct node *p,*q;
	printf("Enter the value:\n");
	scanf("%d",&val);
	q=dl;
	while(q->next!=NULL&&q->data!=val)
	{
		q=q->next;
	}
	q->pre->next=q->next;
	q->next->pre=q->pre;
	free(q);
	return dl;
}
struct node * delete_pos(struct node *dl)
{
	dl=create_dl(dl);
	struct node *q;
	int i,pos;
	printf("Enter position:\n");
	scanf("%d",&pos);
	q=dl;
	for(i=0;i<pos-1;i++)
```

```
    {
        q=q->next;
    }
    q->pre->next=q->next;
    q->next->pre=q->pre;
    free(q);
    return dl;
}
```