```c
#include<stdio.h>

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

typedef struct node * nodeptr;
nodeptr p,list,temp;

nodeptr getnode();
void insert_left();
void insert_last();
void delete_left();
void delete_last();
void display();


void main()
{
int c,i,x;
char ch;
do
{
printf("Menu\n");
printf("1. Insert_left\n 2. Delete_left\n 3. Insert_last\n 4. Delete_last\n
5. display\n");
scanf("%d",&c);
switch(c)
{
case 1: scanf("%d",&x
      push(x);
      break;
case 2: pop();
      break;
case 3: scanf("%d",&x
```

```c
        insert_last(x);
        break;
case 4: delete_last();
        break;

case 5: display();
        break;
}
printf("\ndo you want to continue\n");
scanf(" %c",&ch);
}while(ch!='n');
}

nodeptr getnode()
 {
 nodeptr p;
 p=(nodeptr)malloc(sizeof(struct node));
 return p;
 }

void push(int x)
 {
    temp=getnode();
   temp->info=x;
   temp->next=list;
   list=temp;
   return;
}

void pop()
{
int x;
if(list==NULL)
{
printf("deletion not possible. No nodes existed\n");
}
else
```

```c
{
p=list;
list=p->next;
p->next=NULL;
printf("delete node info is %d",p->info);
free(p);
}
}
void insert_last(int x)
 { p=getnode();
  p->info=x;
  p->next=NULL;
  if(list == NULL)
    list=p;
  else
   {
   temp=list;
   while(temp->next!=NULL)
    {
    temp=temp->next;
    }
    temp->next=p;
   }
   return;
 }
void delete_last()
{
nodeptr q=NULL;
if (list==NULL)
printf("Invalid deletion\n");
else
{
temp=list;
while(temp->next !=NULL)
{
q=temp;
temp=temp->next;
```

```
}
if (q==null)
list=NULL;
q->next=NULL;
printf("%d", temp->info);
free(temp);
}
Return;
}




void display()
{
p=list;
if(p==NULL)
{
printf("No nodes existed\n");
}
else{
while(p!=NULL)
{
printf("%d\t",p->info);
p=p->next;
}
}
}
```