# Doubly.h

```
struct node
{
int data;
struct node *next,*prev;
};
struct node *head=NULL,*ptr=NULL,*ptr1=NULL;
struct node *create()
{
struct node *ptr=NULL,*nw=NULL;
int n,i;
printf("\n How many values : ");
scanf("%d",&n);
printf("\n Enter actual value : ");
for(i=0;i<n;i++)
{
nw=(struct node*)malloc(sizeof(struct node));
scanf("%d",&nw->data);
nw->next=nw->prev=NULL;
if(head==NULL)
ptr=head=nw;
else
{
ptr->next=nw;
nw->prev=ptr;
ptr=ptr->next;
}
}
return (head);
}
void display()
{
struct node *ptr=NULL,*ptr1=NULL;
ptr=head;
do
{
printf("%d",ptr->data);
ptr1=ptr;
ptr=ptr->next;
}
while(ptr!=NULL);
//do
//{
//printf("%d",ptr1->data);
//ptr1=ptr1->prev;
//}
//while(ptr!=NULL);
}

void insert()
```

```
{
struct node *nw=NULL,*ptr=NULL,*ptr1=NULL;
int pos,count=1,x;
printf(" \n Enter position : ");
scanf("%d",&pos);
ptr=head;
nw=(struct node *)malloc(sizeof(struct node));
printf("\n Enter the value to insert : ");
scanf("%d",&x);
nw->data=x;
nw->next=NULL;
if(pos==1)
{
nw->next=head;
head=nw;
}
else
{
while(count!=pos-1)
{
ptr=ptr->next;
count++;
}nw->next=ptr->next;
ptr->next=nw;
}
}
struct node *delet()
{
struct node *ptr=NULL,*ptr1=NULL,*temp=NULL;
int i,pos;
printf("\n Enter position : ");
scanf("%d",&pos);
if(pos==1)
{
temp=head;
head=head->next;
head->prev=NULL;
}
else
{
for(ptr=head,i=1;i<pos-1 && ptr!=NULL;i++)
{
ptr=ptr->next;
}
if(ptr!=NULL)
{
temp=ptr->next;
if(temp->next!=NULL)
{
ptr1=temp->next;
ptr->next=ptr1;
ptr->prev=ptr;
}
else
{
ptr->next=NULL;
printf("The deleted element is %d",temp->data);
}}}}
void search()
{
int key,flag=0,pos=1;
printf("Enter value to be searched : ");
scanf("%d",&key);
ptr=head;
do
{
if(ptr->data==key)
{
pos++;
flag=1;
break;
}
else
ptr=ptr->next;
}while(ptr!=head);
if(flag==1)
{
printf("\n Element is found at position %d",pos);
}
else
printf("\nElement not found");
}
```